

Міністерство освіти і науки України  
Національний університет «Одеська політехніка»  
Інститут інформаційної безпеки, радіоелектроніки та телекомунікацій  
Кафедра кібербезпеки та програмного забезпечення

Балан Катерина Юріївна,  
студентка групи РФ-181

## **КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА**

Автоматизація планування навчальної роботи кафедри. Формування  
навчального плану освітньої програми

Спеціальність:

122 Комп'ютерні науки

Спеціалізація, освітня програма:

Програмне забезпечення систем захисту інформації

Керівник:

Лебедєва Олена Юріївна,

к.т.н., доцент

Одеса – 2022

Міністерство освіти і науки України  
Національний університет «Одеська політехніка»  
Інститут інформаційної безпеки, радіоелектроніки та телекомунікацій  
Кафедра кібербезпеки та програмного забезпечення  
Рівень вищої освіти перший (бакалаврський)  
Спеціальність 122 – Комп'ютерні науки  
Освітня програма – Програмне забезпечення систем захисту інформації

ЗАТВЕРДЖУЮ

Завідувач кафедри КБПЗ

\_\_\_\_\_  
д.т.н., проф. А.А.Кобозєва  
\_\_\_\_\_ 202\_р.

## ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

*Балан Катерині Юріївні*

1. Тема роботи: *Автоматизація планування навчальної роботи кафедри. Формування навчального плану освітньої програми, керівник роботи Лебедева Олена Юріївна, к.т.н., доцент, затверджені наказом ректора від „17”.05.2022 р. №168-в .*
2. Зміст роботи: *огляд баз даних та програм для автоматизації роботи вищого навчального закладу, структура бази даних, програмний продукт, охорона праці.*
3. Перелік ілюстративного матеріалу: *інтерфейси існуючих баз даних та програм для автоматизації роботи університетів, діаграма бази даних, яка пропонується для вирішення поставленої мети роботи, знімки екрану програмного продукту.*

#### 4. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
Охорона праці	Доц. Ярова І.А.		

5. Дата видачі завдання “ \_\_\_\_\_ ” \_\_\_\_\_ 2022 р.

#### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання	Примітка
1	<i>Аналіз джерел з теми випускної кваліфікаційної роботи</i>	<i>17-12-2021</i>	<i>виконано</i>
2	<i>Обґрунтування вибору рішення. Збір даних</i>	<i>27-01-2022</i>	<i>виконано</i>
3	<i>Аналіз існуючого програмного забезпечення для планування навчальної роботи кафедри</i>	<i>05-02-2022</i>	<i>виконано</i>
4	<i>Визначення структури та проектування системи</i>	<i>28-02-2022</i>	<i>виконано</i>
5	<i>Програмна реалізація</i>	<i>25-03-2022</i>	<i>виконано</i>
6	<i>Підготовка тексту роботи</i>	<i>10-05-2022</i>	<i>виконано</i>
7	<i>Підготовка презентації та доповіді</i>	<i>24-05-2022</i>	<i>виконано</i>
8	<i>Попередній захист</i>	<i>10-06-2022</i>	<i>виконано</i>
9	<i>Нормоконтроль, рецензування</i>	<i>20-06-2022</i>	<i>виконано</i>

**Здобувач вищої освіти** \_\_\_\_\_

*Балан К. Ю.*

**Керівник роботи** \_\_\_\_\_

*Лебедева О. Ю.*

## ЗАВДАННЯ

на розробку розділу “Охорона праці”

*Балан Катерині Юріївні, група РФ-181*

Інститут інформаційної безпеки, радіоелектроніки та телекомунікацій

Кафедра кібербезпеки та програмного забезпечення

Тема роботи *Автоматизація планування навчальної роботи кафедри.  
Формування навчального плану освітньої програми.*

Зміст розділу:

- 1 Аналіз умов праці і вибір основних заходів виробничої безпеки.
- 2 Аналіз пожежної безпеки і вибір заходів і засобів пожежної безпеки.

Керівник роботи

\_\_\_\_\_ (Лебедева О. Ю.)

« \_\_\_\_ » \_\_\_\_\_ 2022 р.

Консультант з охорони праці

\_\_\_\_\_ (Ярова І.А.)

« \_\_\_\_ » \_\_\_\_\_ 2022 р.

## АНОТАЦІЯ

Кваліфікаційна робота на тему «Автоматизація планування навчальної роботи кафедри. Формування навчального плану освітньої програми» на здобуття першого (бакалаврського) рівня вищої освіти за спеціальністю 122 – Комп’ютерні науки, спеціалізація, освітня програма – Програмне забезпечення систем захисту інформації, містить 19 рисунків, 1 додаток, 12 літературних джерел за переліком посилань. Робота виконана на 69 сторінках загального тексту і 47 сторінках основного тексту.

Метою роботи є розробка системи для планування навчальної роботи кафедри шляхом автоматизації та формування навчального плану освітньої програми.

Проведено аналіз документообігу кафедри та вже існуючих баз даних і програм для автоматизації роботи вищого навчального закладу.

У результаті виконання кваліфікаційної роботи була розроблена програмна реалізація у вигляді CRM-системи, яка зможе полегшити роботу з досить відповідальними та складними документами, а точніше з навчальними та робочими навчальними планами.

## ANNOTATION

Qualification work on the topic "automation of planning of academic work of the Department. Formation of the curriculum of the educational program" for obtaining the first (Bachelor's) level of higher education in specialty 122 computer science, specialization, educational program – software of information security systems, contains 19 Figures, 1 appendix, 12 literary sources on the list of references. The work was completed on 69 pages of general text and 47 pages of main text.

The aim of the work is to develop a system for planning the academic work of the department by automating and forming the curriculum of the educational program.

The analysis of the document flow of the department and existing databases and programs for automating the work of a higher educational institution is carried out.

As a result of the qualification work, a software implementation was developed in the form of a CRM system that can facilitate working with fairly responsible and complex documents, or rather with training and work curricula.

## ЗМІСТ

ВСТУП .....	8
1 ОГЛЯД БАЗ ДАНИХ ТА ПРОГРАМ ДЛЯ АВТОМАТИЗАЦІЇ РОБОТИ ВИЩОГО НАВЧАЛЬНОГО ЗАКЛАДУ .....	10
1.1 Роль баз даних в сучасному суспільстві .....	10
1.2 Огляд програм для автоматизації роботи вищих закладів освіти.....	12
2 СТРУКТУРА БАЗИ ДАНИХ .....	20
2.1 Бази даних та їх призначення .....	20
2.2 Основні поняття баз даних .....	24
2.3 Проектування системи планування навчальної роботи кафедри.....	26
3 ПРОГРАМНА РЕАЛІЗАЦІЯ РОЗРОБЛЕНОЇ СИСТЕМИ.....	31
3.1 Середовище програмування.....	31
3.2 Огляд програмного продукту .....	32
4 ОХОРОНА ПРАЦІ .....	40
ВИСНОВКИ.....	47
ПЕРЕЛІК ПОСИЛАНЬ .....	48
Додаток А. Лістинг програми .....	49

## ВСТУП

У наш час комп'ютерні технології займають досить значну частину життя та у великій мірі допомагають людині у роботі, навчанні та майже у всіх повсякденних справах. Кожного дня та можна сказати що навіть кожної секунди технології йдуть вперед, тим самим полегшують наше життя, адже зі стрімким розвитком технологій стають простішими та більш доступнішими пристрої, програми та багато іншого.

Чимало часу ми витрачаємо на освіту, кожного року висуваються нові вимоги для навчання, адже спеціаліст будь-якого вищого навчального закладу зобов'язаний мати досить високий рівень підготовки. Рівень даної підготовки дуже багато в чому визначається саме програмою навчання вищого навчального закладу. Достатньо головним документом для цього виступають навчальні плани. Досить багато викладачів у своїй роботі стикаються з великою сукупністю документів, а деякі зіштовхуються з доволі важливим, значущим та відповідальним документом – навчальним планом. Для того, щоб скласти навчальний план потрібно витратити чимало зусиль, адже це дуже клопітка та складна робота, у якій доволі легко зробити помилку, так як потрібно самостійно знайти потрібну інформацію, зробити певний підрахунок (наприклад, кількість годин, кредитів і так далі). Крім того, часто документи потрібно змінювати, адже можуть бути нові норми виходячи з яких доводиться вносити певні корективи, а це в свою чергу також є доволі проблематичним, тому для вирішення даних проблем пропонується такий програмний продукт, який буде містити певну базу даних (яка в свою чергу забезпечує зберігання необхідних даних) та у якій буде документообіг кафедри, для уникнення випадкових помилок у розрахунках, виходячи з нових норм, для зручності та багато іншого.

Метою даної кваліфікаційної роботи є розроблення системи для планування навчальної роботи кафедри шляхом автоматизації та формування навчального плану освітньої програми.

Для досягнення поставленої мети необхідно розв'язати наступні задачі:



- аналіз існуючого програмного забезпечення для планування навчальної роботи кафедри;
- визначення структури та проектування системи планування навчальної роботи кафедри;
- програмна реалізація розробленої системи планування навчальної роботи кафедри.

Об'єктом дослідження є системи автоматизації роботи вищих навчальних закладів.

Предметом дослідження являються методи та засоби автоматизації та формування навчального плану освітньої програми в вищих навчальних закладах.

Практична цінність полягає в розробці та реалізації у вигляді програмного додатку системи для планування навчальної роботи кафедри. Програмний додаток має інтуїтивний та простий інтерфейс. Дана програмна реалізація достатньо полегшить роботу та зможе допомогти з необхідністю поновлення, виходячи з нових норм, тим самим підвищить якість навчального процесу.

# 1 ОГЛЯД БАЗ ДАНИХ ТА ПРОГРАМ ДЛЯ АВТОМАТИЗАЦІЇ РОБОТИ ВИЩОГО НАВЧАЛЬНОГО ЗАКЛАДУ

## 1.1 Роль баз даних в сучасному суспільстві

Нині у життєдіяльності людини грає дедалі більшу роль автоматизація, це стосується і автоматизації інформації, адже як вже зазначалось раніше все дуже стрімко змінюється. Знання та деякі дані можуть настільки інтенсивно збільшитись, що друковані видання не будуть здатними швидко зафіксувати та розповсюдити певну інформацію, яка може бути досить важливою. Інформаційне суспільство доволі часто характеризується впровадженням інформаційних технологій у всі сфери життя або ж може визначатись зростанням ролі інформації у соціальних відносинах. Сучасні інформаційні технології можуть надати дуже великі можливості, в особливості вищим навчальним закладам, оскільки це може сприяти підвищенню ефективності навчального процесу та якості підготовки студентів.

Проблеми створення та розвитку баз даних та систем управління базами даних є одними з найважливіших питань протягом усього часу розвитку інформаційних технологій.

Системи управління базами даних розвивалися від розрахованих на одного користувача, які функціонували на одному персональному комп'ютері, розрахованих на багато користувачів – на базі архітектури файл-сервер, потім – на базі архітектури клієнт-сервер і розподілених систем управління базами даних, що функціонують у рамках глобальних мереж.

У даний час, коли йде розвиток комп'ютерних технологій велику популярність зайняло створення баз даних, адже вони складають доволі невід'ємну частину роботи сучасних підприємств, організацій та навчальних закладів. Ще у 1970 роках виникла потреба у доволі швидкому знаходженні або редагуванні конкретної інформації, оскільки постала проблема у неможливості ручного опрацювання через збільшення кількості даних. Саме у той час компанія ІВМ почала розробляти мови запитів, а саме SQL та QBE, які навіть на

сьогоднішній день реалізовані у дуже багатьох системах керування базами даних. Мова SQL є доволі простою, у наш час вона користується попитом, адже її вагомою перевагою є те, що вона не зобов'язує вказання алгоритму отримання даних, тому дану мову запитів можна з легкістю використовувати при написанні будь-якого програмного коду, і це дає широкі можливості програмістам у створенні або модифікації баз даних та вибірці певної конкретизованої інформації з них у режимі реального часу [1].

Бази даних сьогодні є основою побудови великої кількості інформаційних систем і застосовуються для автоматизації практично всіх сфер людської діяльності.

Автоматизація професійної діяльності визначається як процес створення, впровадження та використання технічних, програмних засобів та математичних методів, що звільняють людину від безпосередньої участі в отриманні, перетворенні та передачі енергії, матеріалів та (або) інформації у професійній діяльності.

Бази даних можуть застосовуватися у сфері органів влади та управління у документальному вигляді, у промисловій, сільськогосподарській, будівельній сферах, в освіті та навчальних закладах. БД застосовують у торгової, фінансової сферах. Бази даних знайшли широке застосування у зберіганні соціальних даних: довідники, освітні установи, розклади літаків та поїздів, особиста інформація про співробітників організацій та багато інших.

Для освітніх установ актуальне використання журналу. У ньому може зберігатися повний спектр інформації про учнів (ПІБ, дата народження, адреса, ...) та перелік занять (роботи, предмети та ін.). Для створення таких баз даних використовують засоби управління базами даних (СУБД). СУБД – це комплекс технічних та програмних засобів, призначених для організації взаємодії БД та користувача. У таких системах автоматизовані процеси, необхідні для роботи з БД.

Система обробки даних (СОД) орієнтована оновлення інформації. Прикладом таких систем може бути інформаційна система заробітної плати

установи. Вона містить інформацію про кількість відпрацьованих годин викладачів, заробітну плату, розрахунок надбавок, підбиття підсумків за період роботи тощо.

За допомогою СУБД також можна розробити каталог бібліотечного фонду освітньої установи. Це дозволить спростити роботу співробітників бібліотеки при оформленні документів на замовлення та видачу необхідної літератури, викладачів та студентів у пошуку необхідного або близького за темою матеріалу, а також пошуку за певним критерієм (автор, рік видання та ін.) та багато іншого.

Можна зробити висновок що бази даних займають важливу роль у сучасному суспільстві, адже на сьогоднішній день без їх використання не можливо обійтись, оскільки завжди є прагнення оброблювати досить великі обсяги інформації та вирішувати певні завдання комплексного характеру. За доволі короткий проміжок часу, а саме кілька десятиліть розвитку бази даних стали одним з дуже важливих і ключових елементів та дають великі можливості в обробці даних.

Впровадження автоматизації систем для вищих навчальних закладів є дуже актуальним. Наразі знайти таку програму саме для кафедри є неможливим, тому дана автоматизація є дуже актуальною задачею.

## 1.2 Огляд програм для автоматизації роботи вищих закладів освіти

Загалом аналізуючи які є бази даних та програми автоматизації роботи вищого навчального закладу можна розглянути наступні:

- програми-планувальники, так як часто є обмеження у часі, тому для оптимізації роботи кафедр доцільно впроваджувати дані програми (наприклад, Microsoft OneNote, Google Keep).
- бази даних, для того щоб вирішити проблему складності здійснення оперативного пошуку, старіння або втрати деяких важливих даних зі зростанням обсягів документованої інформації (наприклад, phpMyAdmin, Access).
- програми для автоматизованого створення розкладу для вищого

навчального закладу (наприклад aSc TimeTables).

- програма Excel допомагає у складанні або обчисленні багатьох документів.

Розглянемо більш детально кожний пункт окремо.

Microsoft OneNote, Google Keep дані програми є дуже корисними та потужними інструментами по самоорганізації роботи викладачів, адже дають наступні можливості:

- бачити всі операції в поточному режимі;
- дозволяють спланувати робочий день;
- керувати певними проектами;
- зберігати необхідні документи на хмарних серверах;
- синхронізувати доступ між усіма пристроями, що є дуже зручним;
- аналізувати завдання будь-якої складності.

Також є дуже зручним, що можна встановити дані програми на будь-який пристрій та синхронізувати між ними, якою саме користуватись програмою це залежить якраз від зручності користувачів та виходячи з завдань, які потрібно щоб програма вирішувала. Якщо порівнювати Microsoft OneNote і Google Keep можна сказати наступне, що кожен з них має свої як позитивні так і негативні сторони. Google Keep має нагадування, які можуть базуватись на місці знаходженні, може надсилати нагадування, має функцію пошуку, термін виконання певного завдання, може архівувати старі дані [3] (рисунок 1.1).

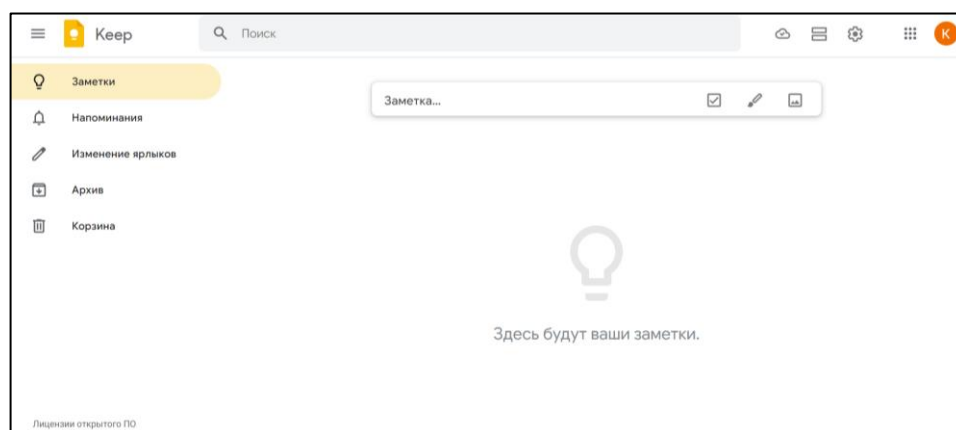


Рисунок 1.1 – Інтерфейс програми Google Keep

One Note може організувати робочі матеріали, має функцію сортування завдань, можна поставити статус завданню, приєднувати певні файли, дана програма зручна для використання колективної роботи (рисунок 1.2).

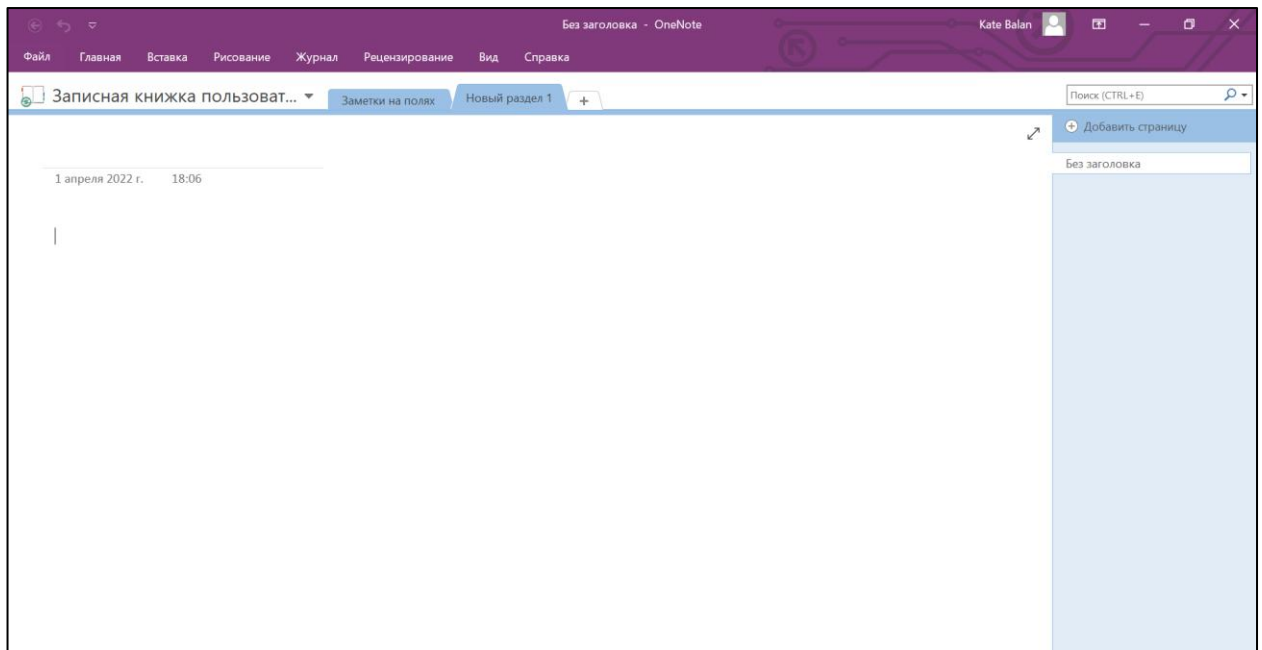


Рисунок 1.2 – Інтерфейс програми One Note

Як вже було зазначено вище існують певні бази даних, для того щоб вирішити проблему складності здійснення оперативного пошуку, старіння або втрати деяких важливих даних, так як частіше за все зростають обсяги документованої інформації. Розглянемо таку програму як phpMyAdmin.

PhpMyAdmin є досить популярним веб-додатком для роботи з базами даних і таблицями (рисунок 1.3), він виступає певною графічною оболонкою, яка дозволяє доволі комфортно і зручно працювати з базою даних, а саме:

- додавати конкретні таблиці;
- наповнювати певними значеннями;
- видаляти деякі або всі таблиці та значення;
- оновлювати всі ці значення;

- копіювати;
- перейменовувати;
- завантажувати текстові файли у таблиці;
- експортувати дані у багатьох форматах;
- створювати графічну схему бази даних у форматі PDF;
- здійснювати пошук в окремих розділах або у всій базі даних;
- модифікувати збережені дані у різні формати;
- можливість писати запити на SQL для виконання різного плану завдань.

Працювати з phpMyAdmin неважко, можна використовувати локальний сервер, або віддалений сервер. [4]

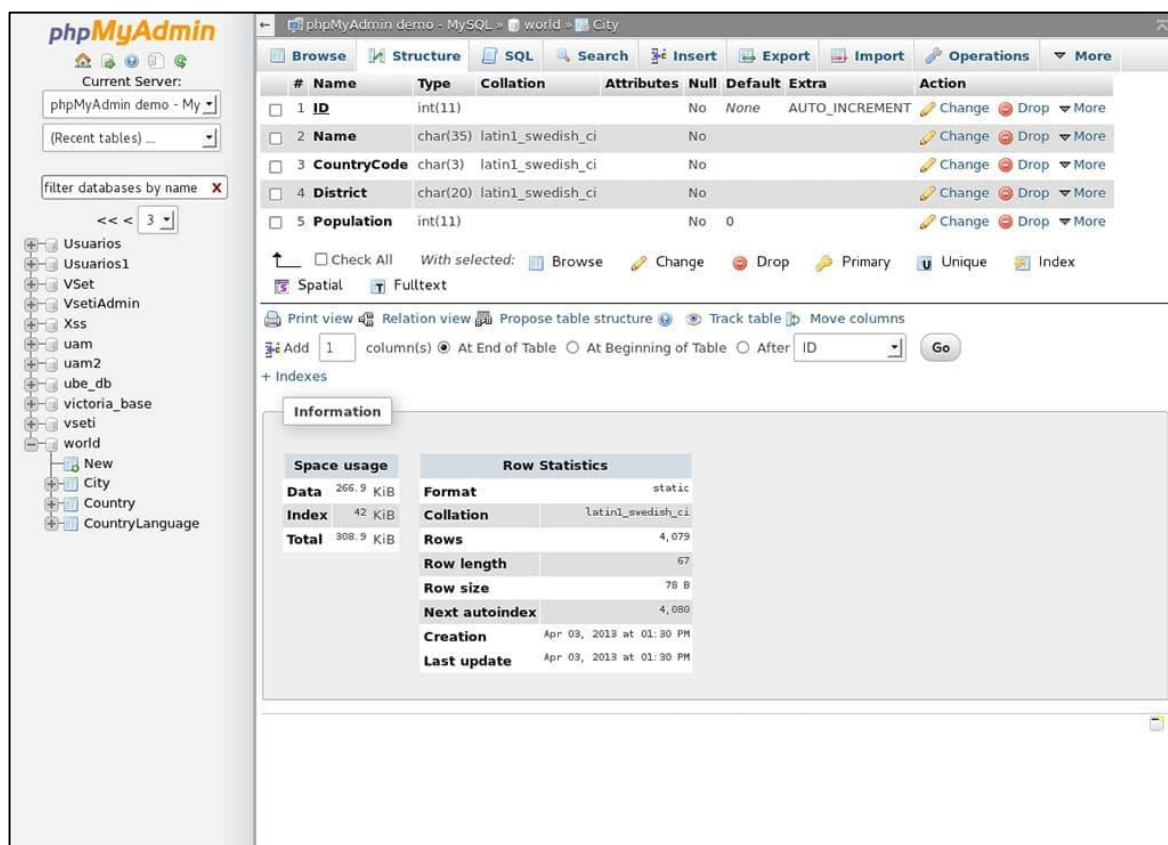


Рисунок 1.3 – Інтерфейс програми phpMyAdmin

Під будь-якою базою даних розуміють сукупність пов'язаною між собою інформацією, яка об'єднана разом у вигляді таблиць. Основу будь-якої бази даних становлять двовимірні таблиці. Наприклад, якщо створити таблицю «викладачі»,

рядки цієї таблиці виступає записом, а стовпці – полями. Для взаємодії користувачів з базами даних використовується система управління даними. Однією з дуже поширених систем управління базами даних є Access [2]. Дана програма при обробці інформації розглядає базу даних як набір декількох структурних елементів, кожен з яких може включати один або декілька об'єктів. Основними складовими будь-якої бази даних у Microsoft Access є наступне:

- таблиця;
- запит (призначений для отримання необхідних даних з таблиць, які є у базі даних);
- форми (використовуються для обробки інтерфейсу, за допомогою якого відбувається введення даних, а також відображення на екрані наявної у базі даних інформації);
- звіт (використовується для підбиття будь-яких підсумків на підставі наявних даних і виведення цих підсумків у певному форматі на друк);
- макроси (потрібні для виконання певних дій при виникненні тої чи іншої події, за їх допомогою можна створювати запити, формувати звіти, відкривати таблиці і виконувати інші дії над об'єктами бази даних) [5].

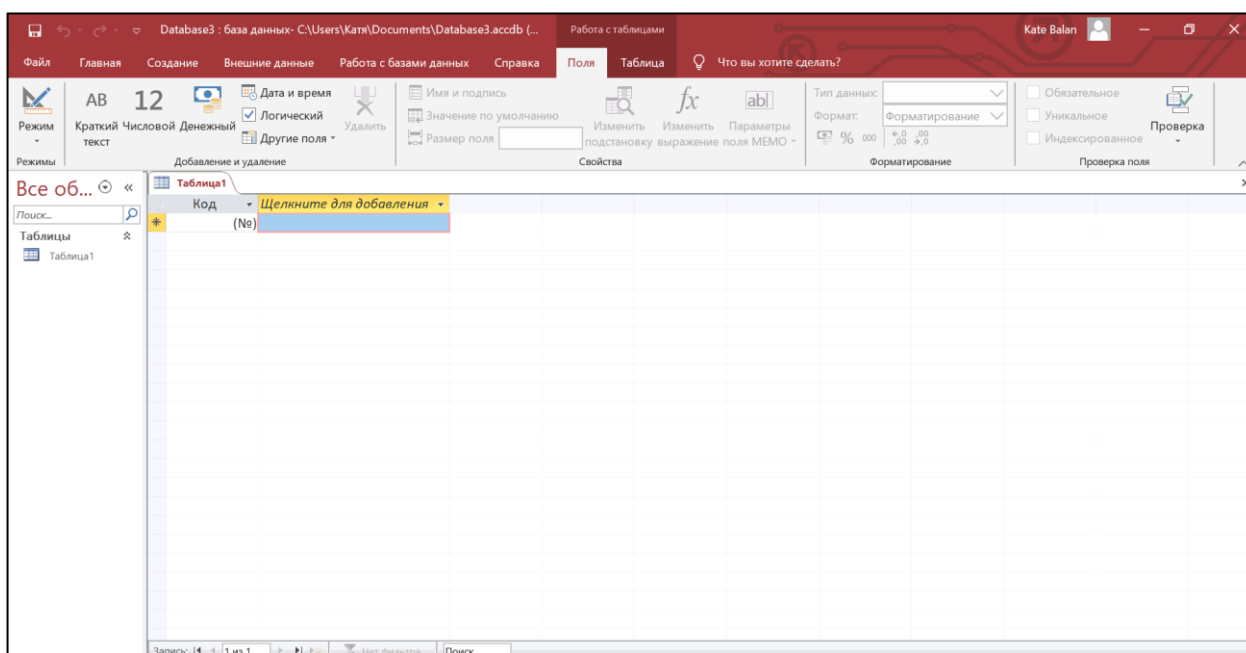


Рисунок 1.4 – Интерфейс программы Access



Також існують певні програми для автоматизації роботи вищого навчального закладу, а саме для автоматизованого створення розкладу, наприклад aSc TimeTables. Це програма, яка дає змогу досить швидко та зручно створити розклад, вказуючи аудиторії та викладачів. Ще доволі зручно що у даній програмі можна вказувати тижневі ліміти часу на кожного викладача і aSc TimeTables може навіть повідомити людину, яка складає даний розклад про те чи є можливість поставити певний предмет на конкретну годину та чи вільний викладач у заданий день і час. Тобто у дану програму потрібно ввести вимоги до розкладу, які потрібні для того чи іншого вищого навчального закладу. Потім програма сама генерує оптимальний розклад, так як всі критерії викладачів враховані, ефективність навчального процесу збільшується в рази. ASc TimeTables проста програма, яка дуже економить час та може створити підходящий розклад. Доступ до розкладу можуть отримати як викладачі так і студенти. Також у ній можна ввести парні і непарні тижні, визначити суботу як робочий день та задати необов'язкові предмети [6].

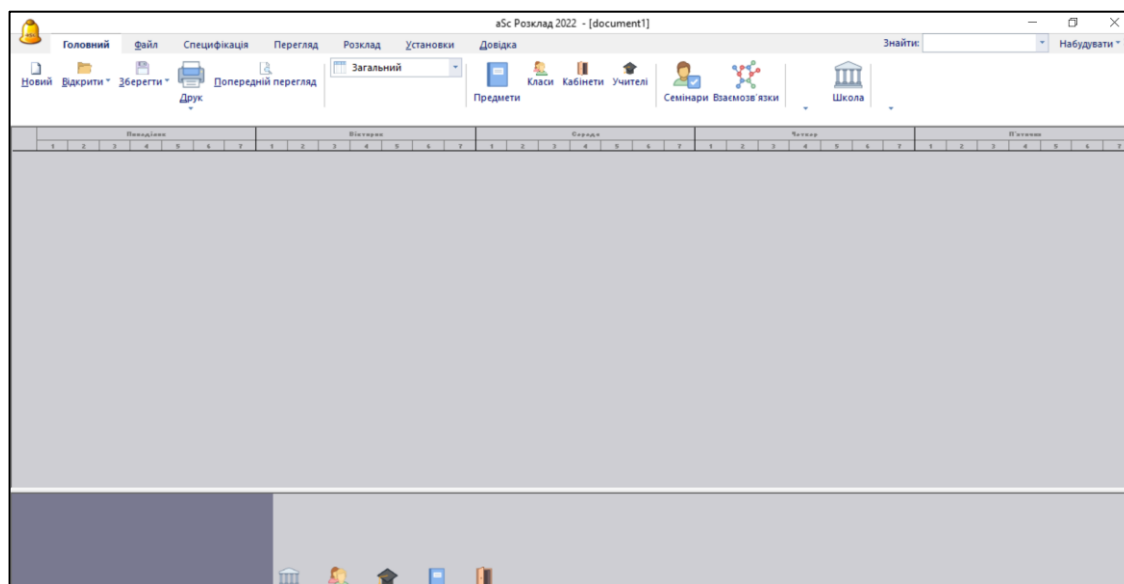


Рисунок 1.5 – Інтерфейс програми aSc TimeTables

Розглянемо програму Excel. Дана програма є, мабуть, найвідомішою і

популярною з категорії електронних таблиць. Excel став своєрідним стандартом в індустрії; одним з його основних переваг є універсальність. Хоча є деякі загальноприйняті сфери, в яких використовується Excel, проте можливості даної програми якщо і є не безмежними, але дуже широкими, наприклад:

- виконання розрахунків майже будь-якої складності;
- автоматизація подібних між собою розрахунків;
- створення звітів;
- організація певних списків та переліків;
- аналіз та обробка числових даних;
- створення діаграм і графіків на основі певних даних;
- створення малюнків та схем;
- можливість за допомогою формул або деяких функцій виконувати обчислення [7].

Тобто, Excel досить зручна програма та може бути використана для автоматизації деякої роботи у вищих навчальних закладах, також вона є досить нескладною.

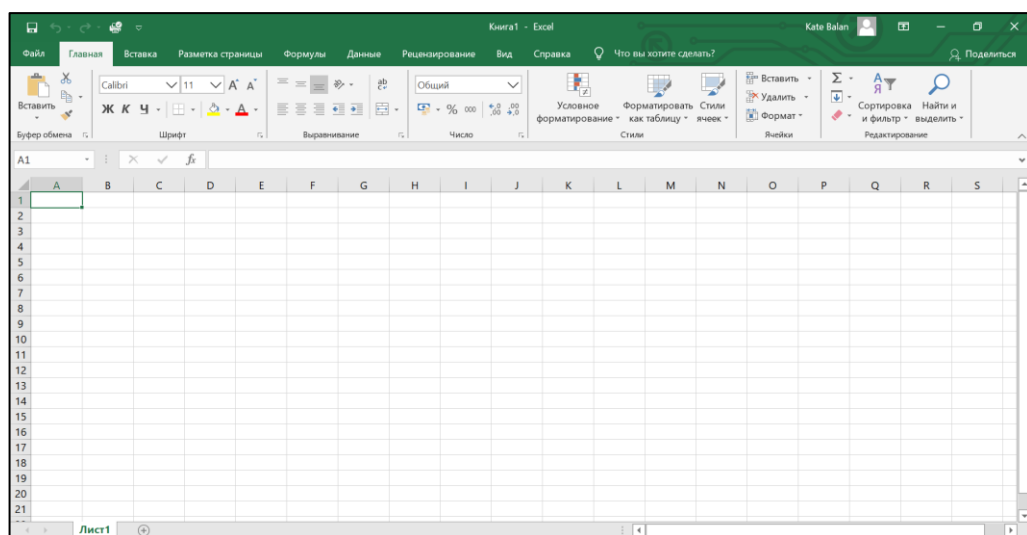


Рисунок 1.6 – Интерфейс программы Excel

Підсумовуючи та аналізуючи програми, які були описані вище, на жаль, можна зробити висновок що не знайшлось такої програми, яка б не тільки містила

у собі базу даних кафедри, а й у якій можна було б створювати навчальні та робочі навчальні плани, також у якій була б можливість вносити певні корективи, виходячи з нових норм, яка була б зручною для використання і так далі.

## 2 СТРУКТУРА БАЗИ ДАНИХ

### 2.1 Бази даних та їх призначення

Вже з назви може стати зрозумілим що бази даних використовують для зберігання певної інформації, тобто деяких даних. Якщо є велика кількість деякої інформації, можна навести приклад з соціальною мережею, ми знаємо що у ній зберігаються імена великої кількості користувачів, паролі, які використовуються як засіб захисту та виступають також певним ідентифікатором, також зберігаються певна сукупність чатів, в яких є певний користувач та багато іншого, все це і знаходиться у базі даних. Звичайно дуже часто використовують для запису певної інформації або деяких даних такий програмний продукт як Excel, але потрібно пам'ятати що таблиця в Excel це файл, і якщо зберігати всю інформацію у файлах, то велика ймовірність того що можна дещо загубити, а це може виявитись дуже цінною інформацією.

Безумовно у минулому програмісти зберігали всю інформацію у текстових файлах, але зрозуміло що чим більше інформації потрібно зберігати, тим складніше було працювати з даними. Для того щоб не загубити інформацію, її потрібно структурувати, при цьому щоб було зручно записувати, отримувати, змінювати та щоб все було безпечно. Так і з'явилися реляційні бази даних. Коротко кажучи реляційні бази даних це таблиці, які зв'язані між собою. Наприклад, якщо припустити таку ситуацію що потрібно записати всіх людей з роботи у певну базу даних. Для цього потрібно :

- створити таблицю, у якій кожен співробітник має свій id, тобто унікальний ідентифікатор;
- у даній таблиці заповнюємо ПІБ, посаду, номер телефону та іншу інформацію, яку потрібно зберігати для співробітників;
- створюємо другу таблицю, у якій заповнюємо всі відділи, які є на роботі, кожен відділ буде мати також свій id;
- заповнюємо всі назви відділів у таблицю, деякий опис кожного відділу, кількість співробітників та іншу інформацію, яка потрібна;

- далі додаємо у першу таблицю з співробітникам стовпчик з відділами, який буде називатись так само як і друга таблиця (з відділами), та заповнюємо ті id відділів, які відносяться до певного співробітника.

Таким чином, можна дізнатись до якого відділу відноситься той чи інший співробітник. Приблизно таким чином влаштовані реляційні бази даних, але звичайно інформації, таблиць та даних в рази більше. Певна річ що існує дуже велика кількість моделей подання даних, наприклад, крім реляційної ще існує:

- ієрархічна;
- «ключ - значення»;
- «пошукова система»;
- документо-орієнтовані;
- мережева;
- об'єктна, тощо.

Розглянемо трохи детальніше кожен з даних пунктів. Ієрархічні бази даних вважаються першими, яких створювали, якщо розглядати зі сторони вже структурованих даних. У ієрархічних баз даних є певна ієрархія об'єктів, у яких один об'єкт залежить від іншого. Прикладом цього може бути навіть файлова система операційної системи Windows, у якій можна наглядно побачити що кожен об'єкт є в підпорядкуванні що найбільше одного об'єкта. Проблема даної бази даних може виникнути коли потрібно щоб зв'язки були не у вигляді дерева, а у вигляді іншої структури, тобто коли певний об'єкт може мати двох або трьох «батьків» і так далі, але перевагою є легкість представлення даних з ієрархічною природою. Ієрархічні бази даних були популярні у 1960-тих роках, на їх зміну прийшли мережеві, які вже могли підтримувати кілька батьківських елементів для кожного об'єкту. Відповідно обидві типи бази даних якщо звернутись до теорії алгоритмів вони пов'язані з графовими структурами, ієрархічна база даних це дерево, мережева база даних це граф, тобто алгоритми які передбачені в цих типах баз даних базуються на складних алгоритмічних структурах.

Для того щоб працювати з базами даних були придумані системи управління базами даних, саме за допомогою цих систем можна отримувати,

оновлювати, впорядковувати та оптимізувати інформацію та багато іншого. Також на що треба звернути достатньо уваги, що за допомогою даних систем можна резервувати та відновлювати базу даних.

Що стосується мережевих баз даних можна виділити наступну перевагу це простоту побудови довільних зв'язків, але проблемою виникає складність представлення даних на фізичному носії і дана складність лежить на користувачеві. Дана проблема виникає з наступного, наприклад у ієрархічній бази даних є вкладені каталоги, у мережевих вже потрібні якісь посилання або реалізовувати у вигляді складних текстових файлів і відповідно базувати на цьому певні алгоритми. Ще одним недоліком є недостатність декларативних засобів пошуку, тобто через те що бази даних у вигляді графу мають певні принципи збереження доволі складно побудувати якусь загальну мову, яка буде відповідати на такі запити. Саме тому мережеві бази даних не є популярними на даний момент, хоча використовуються у певних вузькогалузевих задачах, у яких потрібні саме такі структури.

На зміну мережевим базам даних прийшли реляційні бази даних. Виходячи з вище сказаного про реляційні бази даних можна сказати наступне, що за основу в них береться дуже простий об'єкт – таблиці, та як з ними працювали всі, це привело до того що реляційні бази даних здобули доволі швидке прийняття та дуже стрімку популярність. Звичайно перевагою виступає: простота представлення даних у вигляді таблиць та збереження. Але звісно є недолік що з плином часу може бути повільна робота через велику кількість перевірок цілісності. Взагалі це є недоліком всіх типів баз даних, тому що дана перевірка займає звісно час, якщо цих перевірок багато або вона задана для стовпчика, який має дуже багато значень, то відповідно це займатиме певний час. Також коли у реляційних баз даних багато зв'язків і потрібно отримувати дані з кількох таблиць одночасно це теж забирає деякий час. Звичайно виникає питання чому реляційні бази даних не стали останніми базами даних, адже у них є багато переваг. Справа у тому що реляційна база даних, як вже було зазначено раніше зберігає дані у таблиці, а сучасні інформаційні системи в мові програмування оперують зазвичай

об'єктами. Тому постало питання збереження інформації у базі даних так само як вона зберігається в програмі. Таким чином з'явилась об'єктна база даних.

Дана база даних бере за основу об'єктно-орієнтоване програмування. Об'єктна база даних представляє дані у тому ж вигляді, що і використовується в програмному забезпеченні, це є дуже простим, адже не виникає необхідності писати якісь перетворення об'єктів у таблиці. Потрібно зазначити що можливість об'єктно-орієнтованого програмування тут дуже добре використовується, тобто може бути певне наслідування. Але об'єктні бази даних не набули такого поширення, як наприклад реляційні бази даних через те що все ж таки робота з об'єктами та об'єктно-орієнтованим програмуванням передбачає певну базу, яку потрібно отримувати, а робота з таблицями є трохи простішою.

Наступний тип баз даних, який з'явився досить нещодавно це документо-орієнтовані бази даних. Як вже стало зрозумілим, що в деяких випадках реляційні бази даних не є ефективними, наприклад коли потрібно узагальнити дані з дуже багатьох таблиць одночасно. Перевагою розглядаємої бази даних є те що дані вже готові для виведення. Якщо даний запит виконується постійно, то стає зрозумілим що не має необхідності у здійсненні обробки різних об'єктів. Недоліком є те, що втрачається певна можливість контролю перевірки даних та може виникнути проблема ефективності оновлення додавання даних. Тож документо-орієнтовані бази даних зберігають неструктуровані дані, але мають ускладнене одержання даних з багатьох колекцій, тобто якщо потрібно отримувати дані з багатьох об'єктів, то для цього краще використовувати реляційні бази даних, а якщо потрібно просто зберегти дані, таким чином які вони є і потім їх видобувати, то для цього краще підходять документо-орієнтовані бази даних.

Наступний тип баз даних є відносно новим, з'явилися перші десь у 2003 році, це бази даних типу «ключ - значення». Вони використовуються скоріш не на противагу реляційним базам даних, а в доповнення в певних ситуаціях. Особливість даного типу бази даних полягає в тому що якщо є певний ключ і для нього потрібно отримувати доволі швидко дані, то є така структура як хеш-таблиця, яка дозволяє за ключем отримати дані дуже швидко, це виконується за

використанням хеш-функції. Але проблемою цієї бази даних є у необхідності збереження даних в оперативній пам'яті. Тобто якщо потрібно отримувати дані доволі швидко, то відповідно виникає необхідність збереження в оперативній пам'яті, а на жорсткому диску це вже не так ефективно. Дану базу даних типу «ключ - значення» зазвичай використовують при кешуванні якоїсь інформації, тобто якщо потрібно відповідати на тисячі запитів до одної і тої ж самої таблиці бази даних, то краще їх зберегти десь у базі даних «ключ - значення» і видавати їх доволі швидко з оперативної пам'яті, а не з жорсткого диску, як працює реляційна база даних.

Ще один тип баз даних, який є доволі своєрідним, це бази даних типу «пошукова система». Вони використовуються для здійснення наприклад повного текстового пошуку і зосереджені на збереженні документів та ефективності пошуку. Є зрозумілим що пошук серед текстових файлів є не таким як пошук серед рядків таблиць. Прикладом даної системи є система Google. Особливо ефективні вони для повнотекстового пошуку, але звісно там можуть бути галузеві пошукові системи, які працюють з іншими речами, на кшталт формул, зображень і так далі. Недоліком даної системи є те що вони не ефективні для редагування, тобто дані алгоритми не працюють так добре, у тому випадку якщо потрібно щоб файл оновлювався. [8]

Отже, підсумовуючи вище сказане можна зробити такий невеликий висновок, що база даних це певний додаток, який фіксує: що є певні принципи збереження, перевірки та обробки даних; були розглянуті моделі подання даних (це відповідно саме конкретизовані принципи збереження, оновлення і так далі у вигляді якогось алгоритмічного підходу).

## 2.2 Основні поняття баз даних

База даних – це організована структура, що призначена для зберігання інформації.

База даних створюється для збереження та безпосереднього доступу до інформації, що містить відомості про предметну область. Ступінь конкретизації



даних обумовлюється групою факторів. Насамперед, метою використання інформації з баз даних та складністю інформаційних процесів, що існують у межах предметної області в конкретних умовах.

Система управління базами даних (СУБД) – це програмний механізм, призначений для запису, пошуку, сортування, обробки та друку інформації, що міститься в базі даних.

Структурування – це набір угод про засоби представлення інформації. Тому дані можуть структуруватись по-різному. Розрізняють за структурою мережеву, ієрархічну, реляційну, об'єктно-орієнтовану та гібридну моделі баз даних. Серед усіх цих структур часто використовується реляційна структура.

Розглянемо таке поняття як сутність. Сутність – це множина однотипних об'єктів реального світу, що легко ідентифікується. Зазвичай сутності та зв'язки (це певне відношення однієї сутності до іншої чи до самої себе; зв'язки бувають декількох типів, а саме один-до-одного, один-до-багатьох та багато-до-багатьох) між ними в концептуальному проектуванні позначають на так званих ER-моделях або у моделях «сутність – зв'язок».

У 1976 році Пітером Пін-Шен Ченом був пред'явлений найперший варіант даної моделі, з плином часу багато авторів розробили свої подібні моделі. Слід зазначити що навіть різні програмні засоби, які можуть реалізувати одну й ту ж нотацію можуть відрізнитися своїми можливостями. Сутність на моделях «сутність – зв'язок» позначається прямокутником у центрі якого вказується назва сутності, в овалах навколо певного прямокутника вказуються атрибути (тобто це деяка властивість, яка зображує певну характеристику об'єкта, який описується), сутності, у центрі яких також є назва. Потенційні ключі (дане поняття означає якнайменший набір атрибутів, значення якого точно визначає об'єкт серед усіх об'єктів сутності, яка є заданою) зазвичай на ER-моделях не позначаються, але зазначається первинний ключ (дане поняття означає потенційний ключ, який обраний для ідентифікації об'єкта сутності). Всі атрибути, які входять до первинного ключа підкреслюються лінією.

### 2.3 Проектування системи планування навчальної роботи кафедри

Розглянемо такі поняття як освітньо-професійна програма, навчальний план та робочий навчальний план.

Освітньо-професійна програма (ОПП) є галузевим нормативним документом, у якому визначається нормативний термін та зміст навчання, нормативні форми державної атестації, встановлюються вимоги до змісту, обсягу й рівня освіти та професійної підготовки бакалаврів та магістрів за освітніми програмами.

Навчальний план виступає певним нормативним документом вищого навчального закладу, у ньому регламентується організація навчального процесу з конкретної освітньої програми. Він складається спираючись на відповідну освітньо-професійну програму (яка в свою чергу складається окремо для рівня вищої освіти та кожної спеціальності, та виступає певним нормативним документом, у якому обумовлюється нормативний зміст навчання, встановлюються конкретні вимоги до обсягу, змісту, рівня освіти, у даному документі також є перелік навчальних дисциплін, логічна послідовність їх вивчення, кількість кредитів, потрібних для виконання даної програми, тобто у даному документі описані певні вимоги до рівня освіти). Навчальний план визначає наступне:

- графік навчального процесу;
- кількість кредитів;
- обсяг часу, який передбачений на самостійну роботу, лекції, практичні, лабораторні;
- форму проведення контролю;
- роботи, які передбачені для кожної дисципліни, тощо.

Для досягнення такої мети, як удосконалення змісту навчання, більш конкретне планування навчального процесу, врахування певних конкретних потреб та вимог, своєчасне змінення враховуючи певні накази і розпорядження можуть складатись або коригуватись робочі навчальні плани. Вони також як і навчальні плани складаються окремо для кожної форми навчання та освітньо-

кваліфікаційного рівня.

Робочий навчальний план – це нормативний документ, який розробляється з метою удосконалення змісту навчання, конкретизації планування навчального процесу, своєчасного внесення змін на виконання актів Міністерства освіти і науки України, рішень вченої ради Університету.

Робочий навчальний план складається тільки після утвердження навчального плану. Навчальний план в свою чергу виступає певним «підґрунтям» для робочого навчального плану.

Аналіз предметної області дозволив виділити наступні сутності: дисципліни, кафедри, навчальний рік, кваліфікації, спеціальності, курси, освітня програма, частини плану, цикли, навчальний план та робочий навчальний план. Я реалізувала діаграму, використовуючи нотацію Баркера (оскільки вона є досить поширеною та підходить для даної бази даних), яка складається з сутностей, атрибутів та зв'язків, які потрібні для виконання поставленої мети цієї кваліфікаційної роботи. Розглянемо кожен з них.

Почнемо з сутності «Дисципліни», він має первинний ключ – це код, та такі атрибути: шифр, найменування, батьківський елемент, програма, спеціальність, рівень, частина, цикл, навчальний рік. Кожен навчальний та робочий навчальний план має багато дисциплін, тому зв'язок один-до-багатьох. Дана сутність потрібна для того щоб при створенні навчального плану була можливість «дістати» дисципліни вказуючи програму, спеціальність і так далі.

Сутність «Цикли» має первинний ключ – це код та ще атрибут найменування, він потрібен для навчального плану, оскільки навчальний план складається з багатьох циклів, тому має зв'язок один-до-багатьох.

Сутність «Освітня програма» має первинний ключ – це код та такі атрибути:

- найменування;
- рік початок;
- рік кінець.

Освітня програма вказується у навчальному та робочому навчальному плані тому є зв'язок на діаграмі з цими сутностями.

Сутність «Навчальний план» має первинний ключ – це код, та такі атрибути як: дата, навчальний рік, рівень, спеціальність, програма, форма навчання, кваліфікація, гарант, обов'язкова частина, вибіркова частина, номер рядка, шифр, дисципліна, № семестру, контроль, роботи, кредити, лекції, практичні, лабораторні, самостійні роботи, усього аудиторних та загальний об'єм.

Сутність «Навчальний рік» має первинний ключ – це код, та такі атрибути як: найменування, семестр осінній, семестр весняний. Він зазначається і на навчальному плані і на робочому навчальному плані.

Сутність «Спеціальності» має первинний ключ – код та атрибут найменування, він пов'язаний і з навчальним і з робочим навчальним планом, оскільки у даних планах зазначається спеціальність. Для кожної спеціальності складаються окремі плани, тому зв'язок один-до-одного.

Сутність «Кваліфікації» має первинний ключ – код та такий атрибут як найменування, він використовується в обох навчальних планах.

Сутність «Кафедри» має первинний ключ – це код та наступні атрибути:

- аббревіатура;
- найменування;
- інститут.

Дана сутність потрібна для робочого навчального плану, так як у ньому вказується ще кафедра, оскільки кафедр може бути багато у одному плані, тому зв'язок один-до-багатьох.

Сутність «Курси» має первинний ключ – код та атрибути:

- найменування;
- семестр осінній;
- семестр весняний.

Він також потрібен для робочого навчального плану, зв'язок – один-до-одного.

Сутність «Частини плану» має первинний ключ – код та один атрибут – це найменування. Він зазначається в обох навчальних планах, зазвичай у вищих навчальних закладах частини плану поділяють на обов'язкову та вибірку.

частини. Так як частин плану може бути декілька зв'язок один-до-багатьох.

Сутність «Робочий навчальний план» має первинний ключ – це код, так наступні атрибути: дата, навчальний рік, рівень, спеціальність, програма, форма навчання, кваліфікація, гарант, курс, семестр, обов'язкова частина, вибіркова частина, номер рядка, шифр, дисципліна, № семестру, контроль, роботи, кредити, лекції, практичні, лабораторні, самостійні роботи, усього аудиторних та загальний об'єм.

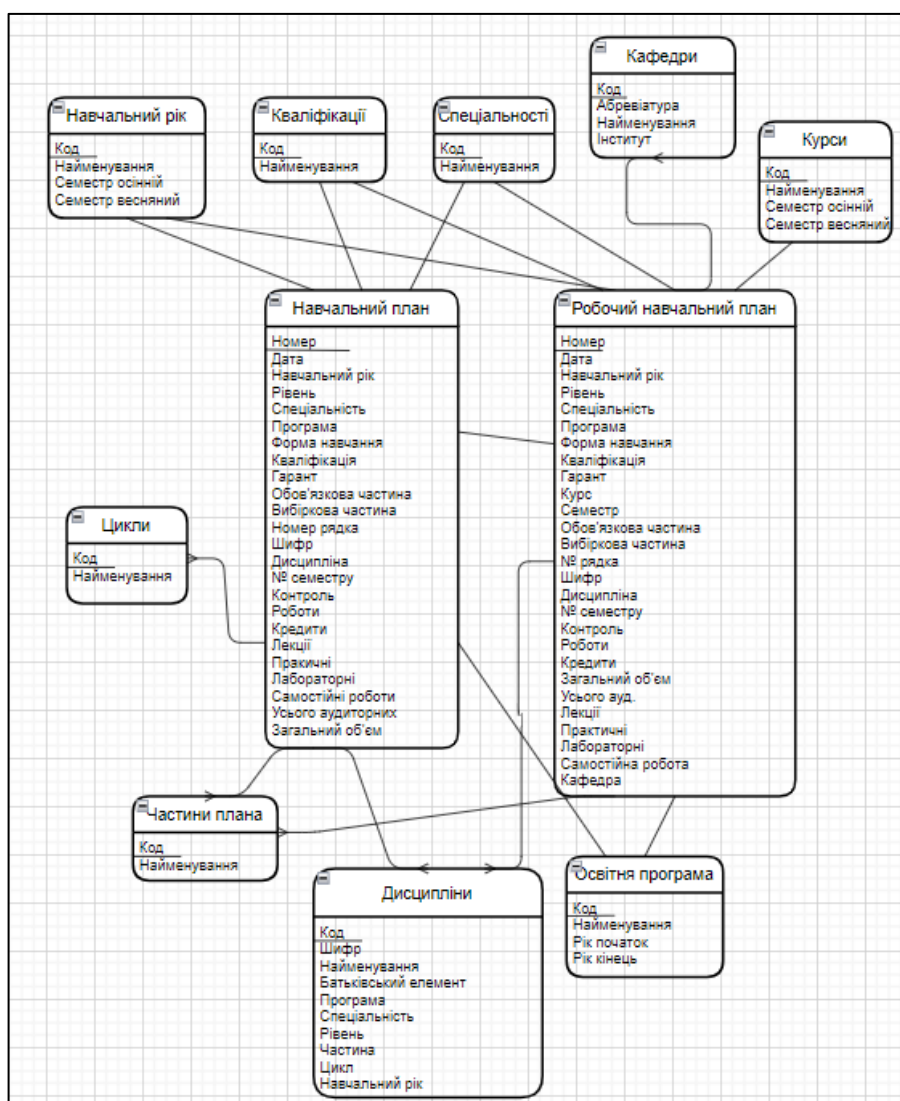


Рисунок 2.1 – Діаграма бази даних, яка пропонується для вирішення поставленої мети роботи

Отже, як вже стає зрозумілим для того щоб скласти навчальний та робочий

навчальний план потрібно витратити дуже багато зусиль, оскільки це є досить відповідальною та складною роботою. У першому розділі даної кваліфікаційної роботи розглядались програми, які використовуються для автоматизації роботи вищого навчального закладу, на превеликий жаль, не знайшлося такої програми, яка була б створена саме для кафедри, яка б мала базу даних, у якій була б також можливість створювати навчальні та робочі навчальні плани, вносити певні корективи, враховуючи нові норми, яка була б зручною для використання, тобто автоматизувати та облегшити дану роботу, у даному розділі було описане поняття бази даних, їх структура та призначення і також була створена та описана діаграма для бази даних, яка представляється для досягнення мети даної кваліфікаційної роботи.

## 3 ПРОГРАМНА РЕАЛІЗАЦІЯ РОЗРОБЛЕНОЇ СИСТЕМИ

### 3.1 Середовище програмування

Почнемо з поняття CRM-системи, вона виступає певною спеціальною програмою, за допомогою якої можна організувати процес взаємодії з клієнтами. Однією з переваг, які є досить важливими для вирішення завдань даної кваліфікаційної роботи є автоматизація, тобто CRM-система допомагає скоротити час та автоматизувати певні однакові задачі, які виконуються постійно. Дана система є універсальною, тобто підходить для різних сфер. Також її перевагою є те, що всі потрібні дані можна зберегти у одному місці та при необхідності відкрити доступ. Використання даних систем відчутно зможе полегшити роботу. CRM-система поділяється на такі види:

- операційні;
- аналітичні;
- колабораційні;
- комбіновані.[9]

У якості CRM-системи для вирішення поставленої мети даної кваліфікаційної роботи використовується 1С. Дана програма систематизує базу даних та дозволяє виконувати за дуже короткий термін певні завдання.

Її перевагами є:

- універсальність;
- багатофункціональність;
- можливість у дуже короткий термін допрацювати програму використовуючи конфігуратор;
- зручність у використанні;
- зберігання всіх даних в одній базі даних;
- можливість проведення резервного копіювання;
- можливість відновлення бази даних;
- надійний захист інформаційної бази. [10]

### 3.2 Огляд програмного продукту

Після запуску розробленої системи з'являється вікно з доступним інтерфейсом, який виглядає наступним чином:

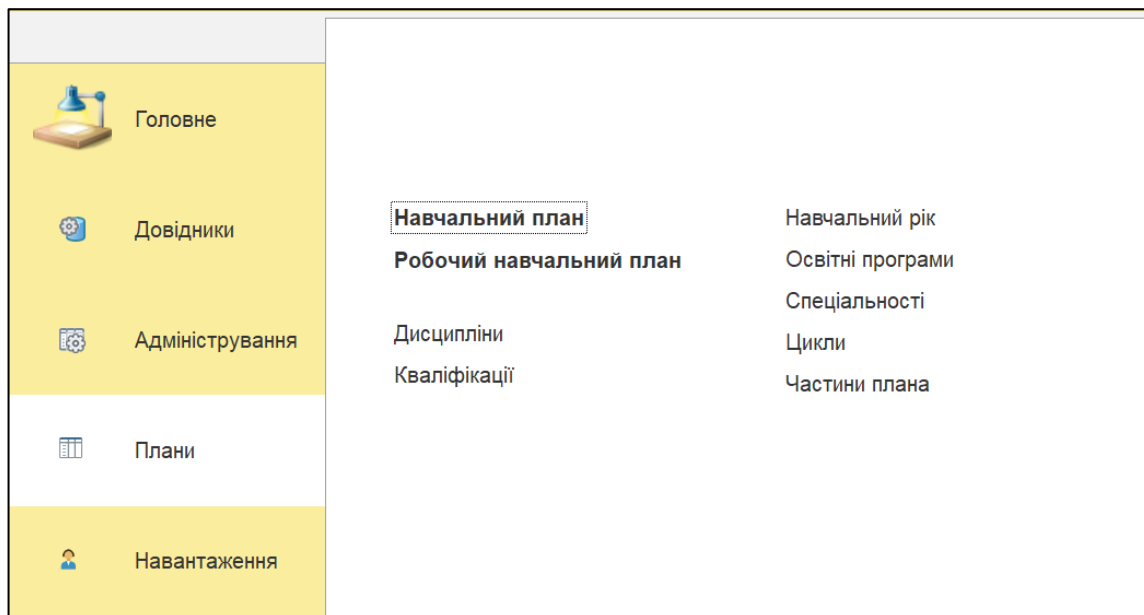


Рисунок 3.1 – Інтерфейс програми

Можна побачити такі підсистеми як адміністрування та плани, виглядають вони наступним чином:

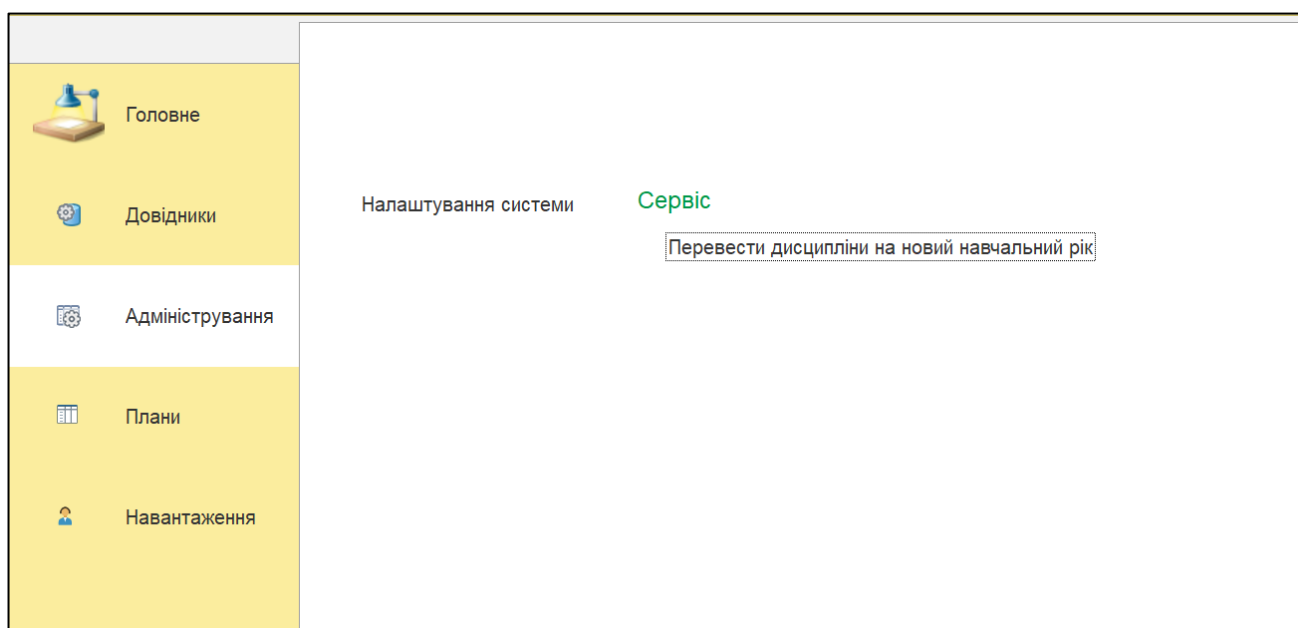


Рисунок 3.2 – Інтерфейс розділу «Адміністрування»



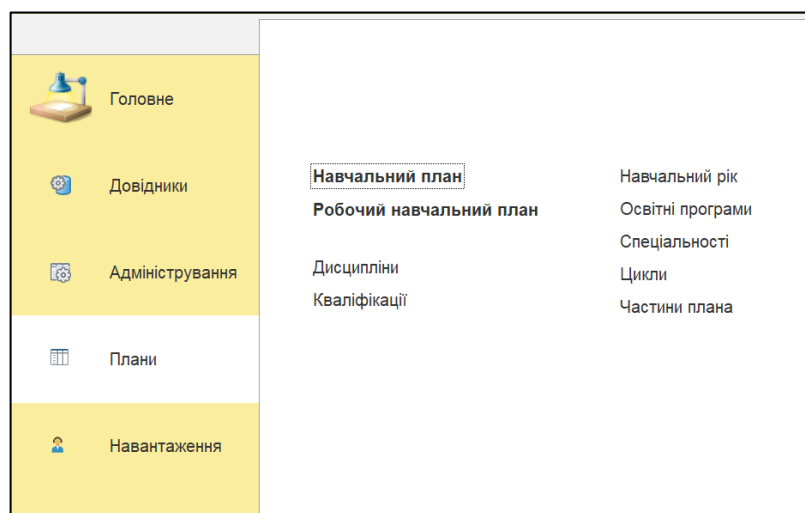


Рисунок 3.2 – Інтерфейс розділу «Плани»

Завданням даної кваліфікаційної роботи є автоматизація та формування навчального та робочого навчального плану. У другому розділі даної кваліфікаційної роботи була розглянута структура даної бази даних, тому розглянемо виконання завдання по створенню нового навчального плану.

Перше, що необхідно у довіднику «Дисципліни» це визначитись зі складом дисциплін. Для цього потрібно перейти у довідник «Дисципліни», виглядає він наступним чином:

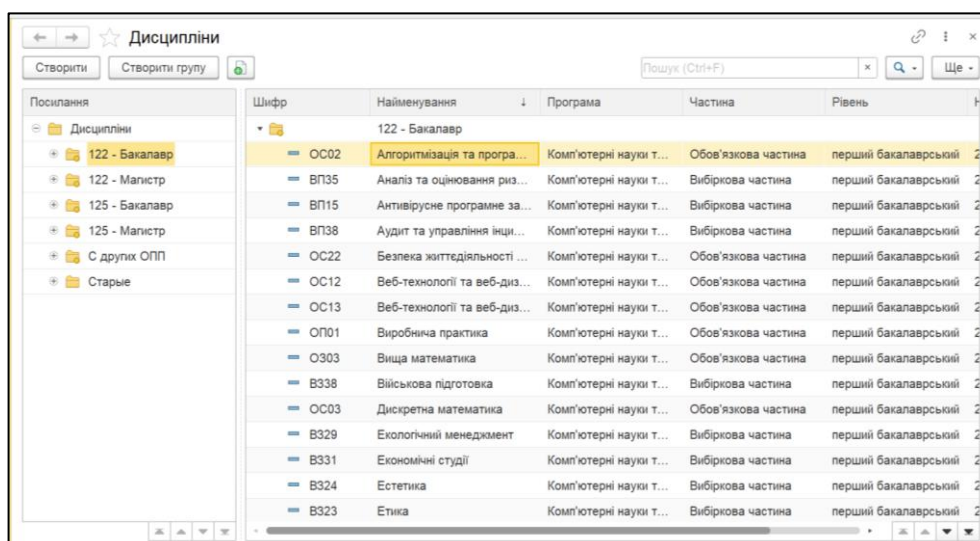


Рисунок 3.3 – Інтерфейс довідника «Дисципліни»

Можна побачити створені папки для зручного пошуку певної дисципліни,

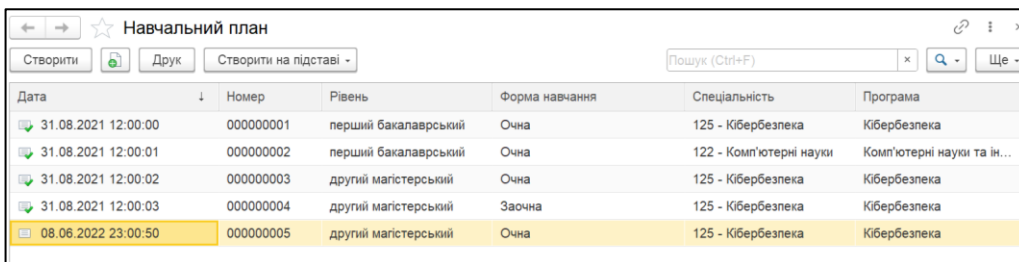
відкриваючи необхідну папку з'являється список дисциплін, які їй належать, можна також створювати нові дисципліни за допомогою кнопки «Створити» або видалити при необхідності, також є можливість легко перенести дисципліну до іншої папки. Крім того, для зручності та швидкості знаходження конкретних дисциплін можна даний довідник трансформувати, для цього потрібно натиснути на кнопку «Ще», далі - «Настроїти список» та у полі «Групування» обрати той критерій, який потрібен для зручності та натиснути «Завершити редагування», таким чином згрупуються дисципліни за тим критерієм, який потрібен.

Для того щоб створити навчальний план наприклад на наступний рік потрібно передивитись всі дисципліни, при необхідності додати або видалити, далі перейти на кожну з них, яка потрібна окремо (для цього необхідно натиснути на неї) та додати той навчальний рік, у якому буде читатись дана дисципліна. Це можна зробити переходячи на кожну дисципліну окремо та додаючи самостійно, натиснувши на кнопку «Додати» або можна перейти у розділ «Адміністрування», далі натиснути на сервіс під назвою «Перевести дисципліни на новий навчальний рік» та заповнити навчальний рік з якого на який потрібно перевести, рівень, програму, спеціальність, форму навчання, кваліфікацію та натискаємо на заповнити, таким чином можна побачити ті дисципліни виходячи з того який рівень, рік, програма і так далі, були вказані, при необхідності є можливість легко видалити або додати дисципліну, далі потрібно натиснути на кнопку «Перевести дисципліни», таким чином буде автоматично доданий той навчальний рік, який був вказаний при заповненні.

Рисунок 3.4 – Інтерфейс сервісу «Перевести дисципліни на новий навчальний рік»

Також у довіднику «Дисципліни» якщо певна дисципліна вже була задіяна у навчальному плані, який був проведений, це можна легко побачити, натиснувши на неї та перейшовши у «Склад дисципліни», у якому є така інформація як: кількість кредитів, навчальний план, у якому вона була задіяна, семестровий контроль, загальний об'єм, кількість годин, номер семестру, форму навчання, кваліфікацію.

Далі для того щоб створити навчальний план потрібно перейти у розділ «Плани» та натиснути на «Навчальний план».



Дата	Номер	Рівень	Форма навчання	Спеціальність	Програма
31.08.2021 12:00:00	000000001	перший бакалаврський	Очна	125 - Кібербезпека	Кібербезпека
31.08.2021 12:00:01	000000002	перший бакалаврський	Очна	122 - Комп'ютерні науки	Комп'ютерні науки та ін...
31.08.2021 12:00:02	000000003	другий магістерський	Очна	125 - Кібербезпека	Кібербезпека
31.08.2021 12:00:03	000000004	другий магістерський	Заочна	125 - Кібербезпека	Кібербезпека
08.06.2022 23:00:50	000000005	другий магістерський	Очна	125 - Кібербезпека	Кібербезпека

Рисунок 3.5 – Інтерфейс списку документів «Навчальний план»

Можна побачити документи, які вже були створені, а для створення нового потрібно натиснути на кнопку «Створити» і далі заповнити навчальний рік, рівень, програму, форму навчання та кваліфікацію. Далі натискаємо на кнопку «Заповнити» та можемо побачити ті дисципліни, які були взяті з довідника «Дисципліни» по тим критеріям, які були заповнені при створенні навчального плану. Далі заповнюємо кількість кредитів та годин на лекції, практичні, лабораторні та самостійну роботу. Таким чином заповнюється обов'язкова та вибіркова частини окремо. (рисунок 3.6, 3.7)

Також є стовпчик під назвою «Усього ауд», який автоматично сумує кількість годин на лекції, практичні та лабораторні, а стовпчик загальний об'єм ще сумує самостійну роботу. Також є певна перевірка, так як кількість годин у кредиті дорівнює 30, то якщо загальний об'єм поділити на дану кількість годин, то отримаємо кількість кредитів. Тобто при заповненні, якщо при ділення не виходить те число, яке було вказано, тоді програма автоматично змінює кількість

годин самостійної роботи та виводить повідомлення, у якому вказується дисципліна, у якій було неправильне співвідношення кредитів та який перерахунок був здійснений.

N	Шифр	Дисципліна	№ семестру	Контроль	Роботи	Кредитів	Лек.	Прак.	Лаб.	СР	Усього ауд.	Загальн
1	О301	Професійна іноземна мова	1 семестр	Залік		3,00		30		60	30	90
2	ОП01	Сучасні КСЗІ та їх особливості	1 семестр	Екзамен	Курсові...	4,50	30	14		91	44	135
3	ОП02	Бізнес-процеси в кібербезпеці	1 семестр	Екзамен	РГР(РР)	4,50	30		14	91	44	135
4	ОП03	Проблеми кібербезпеки та су...	1 семестр	Екзамен	Курсові...	4,50	30		14	91	44	135
5	ОП04	Захист соціо-технічних систем	1 семестр	Залік		3,00	14		16	60	30	90
6	ОП05	Сучасні методи захисту комп'	1 семестр	Екзамен		4,50	30		14	91	44	135
						<b>63,00</b>	<b>180</b>	<b>72</b>	<b>74</b>	<b>1 564</b>	<b>326</b>	

Рисунок 3.6 – Створення навчального плану, інтерфейс обов'язкової частини

N	Шифр	Дисципліна	№ семестру	Контроль	Роботи	Контрольні роботи	Кредитів	Лек.	Прак.	Лаб.	СР
1	В301	Українська мова як іноземна	2 семестр	Залік			4,50		44		91
2	В302	Інтелектуальна власність та ...	2 семестр	Залік			4,50	30	14		91
3	В303	Професійна педагогіка	2 семестр	Залік			4,50	30	14		91
4	В304	Екологічна безпека	2 семестр	Залік			4,50	30	14		91
5	В305	Організаційна психологія	2 семестр	Залік			4,50	30	14		91
6	В306	Патентознавство	2 семестр	Залік			4,50	30	14		91
							<b>123,50</b>	<b>600</b>	<b>998</b>	<b>196</b>	<b>1 911</b>

Рисунок 3.7 – Створення навчального плану, інтерфейс вибіркової частини

У розділі адміністрування є довідник під назвою «Налаштування системи»,

який виглядає наступним чином:

☆ 2022 - 2023 навчальний рік (Налаштування системи)

Записати та закрити    Записати    Ще -

Код: 001    Найменування: 2022 - 2023 навчальний рік

Навчальний рік: 2022 - 2023 навчальний рік

30 Кількість годин у 1 кредиті

Бакалаври    Магістри

Додати    ↑ ↓    Пошук (Ctrl+F)    Ще -

N	Семестр	Кіл-ть тижнів	Макс. аудит.	Кіл-ть іспит.	Кіл-ть вибірк.
1	1 семестр	15	22	4	
2	2 семестр	15	22	4	
3	3 семестр	15	22	4	
4	4 семестр	15	22	4	
5	5 семестр	15	22	4	
6	6 семестр	15	22	4	
7	7 семестр	15	22	4	

Виводити на друк вибіркові дисц. для бакалаврів

Текст для друку вибіркових дисц. для бакалаврів: Каталог вибіркових дисциплін (бакалавр)

Рисунок 3.8 – Інтерфейс «Налаштування системи»

У ньому може заповнюватись по новим нормам кількість годин у 1 кредиті, також є два підрозділи це бакалаври та магістри. Потрібно додавати кожен рядок окремо, заповнюючи семестр, кількість тижнів, максимальне аудиторне навантаження, об'єм лекцій, кількість іспитів та кількість вибіркових дисциплін. Також якщо потрібно у друковану форму виводити весь перелік вибіркових дисциплін, то можна натиснути на «Виводити на друк вибіркові дисц.».

Теж при створенні навчального плану для зручності та перевірки є кнопка «Перевірити», натискаючи на яку створиться нова вкладка, у якій виводиться повідомлення та проміжна таблиця, у якій згортаються всі дисципліни, там можна побачити номер семестру, кредити, кількість аудиторних годин та чи є ця дисципліна вибірковою чи ні. У повідомленні проведений певний аналіз, у якому вказується норма аудиторного навантаження за семестр та розраховується кількість кредитів та кількість годин. Так як у розділі адміністрування у довіднику «Налаштування системи» кількість аудиторного навантаження заповнюється в тиждень, тому щоб отримати за семестр програма розраховує множення максимального аудиторного навантаження на кількість тижнів, таким

чином отримується норма аудиторного навантаження за семестр. Також у даному довіднику заповнюється кількість вибіркових дисциплін, тому у повідомленні кількість кредитів та аудиторних годин розраховується наступним чином: обчислюється сума всіх кредитів за семестр, ділиться на їх кількість, множиться на кількість дисциплін та сумується з тими дисциплінами, які не є вибірковими. Виглядає це наступним чином:

Номер семестру	Кредиты	ВсегоАудиторные	Вибіркова
2 семестр	3,00	30	
2 семестр	4,50	44	✓
2 семестр	4,50	44	✓
2 семестр	4,50	44	✓
2 семестр	4,50	44	✓
2 семестр	4,50	44	✓
2 семестр	4,50	44	✓
2 семестр	4,50	44	✓
2 семестр	4,50	44	✓

Повідомлення:

№ семестра	Кредиты	Аудит.
1 семестр:	Норма аудиторной - 300	
1 семестр:	- 30 - 296	
2 семестр:	Норма аудиторной - 300	
2 семестр:	- 25,5 - 250	
3 семестр:	Норма аудиторной - 9	

Рисунок 3.8 – Інтерфейс перевірки відповідності нормам

Тобто у повідомленні можна побачити яка є норма та порівняти з тими значеннями, які вводились. При остаточному заповненні навчального плану можна натиснути на кнопку «Друк» і створиться друкована форма даного документу, виглядає це наступним чином:

Навчальний план 000000001 від 31.08.2021 12:00:00

НАВЧАЛЬНИЙ ПЛАН

І ГРАФІК НАВЧАЛЬНОГО ПРОЦЕСУ

Строк навчання 3 роки 10 місяців на основі повної загальної середньої освіти

1 семестр 2 семестр

вересень, жовтень, листопад, грудень, січень лютий, березень, квітень, травень, червень, липень, серпень

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49

Т - теоретичне навчання; С - експериментальні сесії; К - курсові; вр - виконання кваліфікаційної роботи; А - вступна (зміст кваліфікаційної роботи, екзамен); П - практична, індивідуальна робота з здобувачами; У - уступова сесія

Рисунок 3.8 – Інтерфейс друкованої форми

Аналогічним чином можна створити робочий навчальний план, але є один момент, що він створюється на підставі навчального плану, тобто дані будуть взяті не з довідника «Дисципліни», а з навчального плану, який був проведений, виглядає він наступним чином:

N	Шифр	Дисципліна	№ семестру	Контроль	Роботи	Кредитів	Загальний об'єм	Усього ауд.	Лек.	Прак.	Лаб.	СР
1	O301	Іноземна мова	1 семестр	Залік		3,00	90	60		60		
2	O302	Історія України та українс...	1 семестр	Екзамен	РГР(РР)	3,00	90	44	30	14		
3	O303	Вища математика	1 семестр	Екзамен		6,00	180	60	30	30		
4	O304	Фізика	1 семестр	Екзамен		6,00	180	76	30	30	16	
5	OC01	Методи та засоби інформ...	1 семестр	Залік		6,00	180	76	30	16	30	
6	OC02	Алгоритмізація та програ...	1 семестр	Екзамен	РГР(РР)	6,00	180	74	30	14	30	

Рисунок 3.8 – Інтерфейс робочого навчального плану

Отже, у даному розділі було проведено:

- аналіз навчального та робочого навчального плану кафедри;
- огляд програмного продукту.

Також були описані дії, які потрібно зробити для створення навчального або робочого навчального плану, враховуючи перевірки у ході створення документів та зручності користувача.

## 4 ОХОРОНА ПРАЦІ

Охорона праці завжди пов'язана до якогось об'єкта, тобто досліджується певний об'єкт на предмет безпеки. Об'єктом дослідження є робоче місце яке оснащено персональним комп'ютером. На робочому місці виникають певні небезпечні та шкідливі виробничі фактори. Як би не намагались працювати завжди є певні негативні впливи, якими не потрібно нехтувати. Дуже важливо зберігати здоров'я, тому потрібно робити все правильно з точки зору охорони праці.

На робочому місці з портативним персональним комп'ютером також є таке обладнання: комп'ютерна миша (як периферійний пристрій), роутер (як елемент мережі Internet), принтер (як периферійний пристрій). Крім того у приміщенні знаходиться: шафа з книгами, стіл, документи, система центрального водяного опалення, суміщена система освітлення (яка складається з штучного та природнього освітлення). Основним обладнанням на робочому місці є ноутбук. Портативний персональний комп'ютер створює наступний небезпечний виробничий фактор: небезпека ураження електричним струмом.

Наприклад, через підвищену напругу в електричній мережі є ризик проходження крізь тіло людини, що може визвати судоми, опіки, непритомність, зупинку дихання та серця, тобто даний фактор є смертельною небезпекою. Є певні петлі струму проходження крізь тіло людини, найбільш небезпечними з них наступні: рука-рука, права нога-ліва рука (або навпаки), голова – будь-яка частина тіла. Величина відчутного порогового змінного струму – 0.6–1.5 мА, постійного – 5-7 мА, невідпускаючого порогового змінного струму – 10-15 мА, постійного – 50-80 мА. Номінальна змінна напруга в електромережах України згідно ГОСТУ 21128-83, на який посилається ГОСТ 13109-97, а також згідно ДСТУ EN 50160:2014 є 220В, яка і є у даному приміщенні. Електромагнітні поля, які характеризуються напруженостями магнітних та електричних полів є найшкідливішими для організму людини. На робочому місці є дисплеї (монітори), що несприятливо впливає на здоров'я через шкідливі випромінювання. Також



може створювати дану небезпеку ураження електричним струмом коротке замикання, випадкове потрапляння води на електроприлади, дотик до струмоведучих частин, що знаходяться під напругою, тобто частіше за все дана небезпека може виникнути внаслідок недотримання правил електробезпеки або виходу з ладу електроприладів.

Що стосується шкідливих виробничих факторів:

- недостатня освітленість робочої зони. Освітлення відіграє дуже важливу роль у житті людини. Близько 90 % інформації сприймається через зоровий канал, тому правильне виконане раціональне освітлення має важливе значення для виконання роботи. Даний фактор несприятливо позначається на зір, згодом і на продуктивність роботи. Незадовільне освітлення на робочому місці може призвести до зниження зорової працездатності. Система освітлення на робочому місці включає в себе штучне та природне освітлення та створює освітленість, яка дорівнює 500 люкс. Оскільки природне освітлення постійно змінюється залежачи від багатьох чинників, наприклад темний час доби, пора року, похмура погода та інше не обійтись без штучного освітлення, для того щоб забезпечити нормовані значення освітлення для даного приміщення. За документом ДБН В.2.5-28-2006 «Природне і штучне освітлення» норма також 500 люкс, тобто фактичне значення освітленості збігається з нормативним.

- підвищений рівень шуму. Шум погіршує умови праці додаючи шкідливу дію на організм людини. На протязі тривалого часу працюючи в умовах постійної шумової дії можуть виникати головні болі, стреси, нервово-психічні напруження, знижується працездатність і продуктивність. Джерелами шуму у даному приміщенні є принтер, кондиціонер, вентиляційні установки та ноутбук. За документом ДСН 3.3.6.037-99 «Санітарні норми виробничого шуму, ультразвуку та інфразвуку» [11] еквівалентний рівень звукового тиску не повинен перевищувати 50 дБА, фактичне значення у даному приміщенні дорівнює 45 дБА. Для того щоб знизити рівень шуму можуть бути облицьовані звукопоглинальними матеріалами стіни і стеля приміщень, де встановлені комп'ютери. Для засобів індивідуального захисту можна використовувати

навушники.

- вимушена робоча поза. Даному шкідливому виробничому фактору варто присвятити також певну увагу, адже тривале перебування у тій самій сидячій позі дуже негативно впливає на хребет. Під час користування комп'ютером необхідно пам'ятати що повороти тулуба, фіксована поза, певні нахили корпусу, все це позначається на здоров'ї. Перебування у певній позі за комп'ютером багато годин поспіль викликає біль у спині, навантаження на міжхребцеві диски, перенапруження м'язів плечового пояса та спини, окрім того неправильна постава призводить до сильного розтягування м'язів, що підтримують хребет. Нормативне значення відстані від монітору до очей 50-60 см. Існують певні рекомендовані комплексні вправи для хребта описані, які призначені для тренування і розслаблення хребта, поліпшують периферійний кровообіг, сприяють збереженню правильної постави та оптимальному перерозподілу м'язового навантаження при роботі. Отже, важливо займатися спортом, стежити за поставою та положенням під час роботи, періодично вставати та виконувати певні рухи або хоча б трохи прогулятися час від часу, але краще за все виконувати спеціальні вправи для хребта.

- дрібні стереотипні рухи пальцями рук. Даний шкідливий виробничий фактор є одним із найпоширеніших, так як проведення багатьох годин поспіль за мишкою і клавіатурою без правильних профілактичних вправ може призвести до певних захворювань. Один з них це тунельний синдром зап'ястя або ще називають синдром зап'ястного каналу, тобто це певне здавлення серединного нерву кисті, при якому відчувається біль, оніміння, погіршення рухливості, поколювання, пальців, втрата тактильних відчуттів. З'являється даний синдром внаслідок постійного навантаження на зап'ястя. Для того щоб не допустити дане захворювання слід по можливості знижувати навантаження, робити профілактичні спеціальні вправи, влаштовувати перерви під час роботи. При виникненні сильного дискомфорту вдягати спеціальну шину, яка підтримує зап'ястя. Також існують певні комплекси вправ для рук, які можна робити у будь-який час протягом дня.

- розумові напруження. Даний шкідливий виробничий фактор залишає за собою певний відбиток психологічного і поведінкового порушення, а саме нервозність, роздратування, тривога, замкнутість і так далі. Спорт є засобом захисту для даного шкідливого фактору. Вміння фокусуватись на одному русі, використовувати достатньо великий асортимент різної рухової активності, вміння керувати своїм тілом, вивчати нові патерни рухів. Також важливо дотримуватись певного режиму харчування та стимулювати розумову діяльність певними продуктами, наприклад банан, горіхи, шматок чорного шоколаду, різнокольорові свіжі овочі, які покращують циркуляцію крові в головному мозку. Існує певний комплекс вправ для поліпшення мозкового кровообігу. Певні нахили та повороти голови, дихальні вправи, тренування вестибулярного апарату підсилює мозковий кровообіг полегшуючи розумову діяльність.

- емоційне напруження. У свою чергу даний шкідливий виробничий фактор призводить до певних додаткових затрат енергії, а саме до виснаження нервової системи, хронічного стресу, який також позначається на продуктивності роботи. Довготривале емоційне напруження може призвести до так званого «вигорання», тому дуже важливо цього не допустити та в цілях профілактики намагатись обдуманно розподіляти свої навантаження та вчитись переключатись з одного виду діяльності на інший. Існують сеанси психофізіологічного розвантаження, які рекомендуються виконувати, а саме прийоми психічної саморегуляції та виконання нескладних фізичних вправ з словесним самонавіюванням.

- параметри мікроклімату. Робота програміста за енерговитратами відноситься до легкої 1а категорії робіт. Фактичні значення у даному приміщенні наступні (теплий період року): температура повітря 23 градуси, відносна вологість – 45, швидкість руху повітря – 0,1 м/сек. Оптимальні величини температури, відносної вологості повітря та швидкості руху повітря в робочій зоні за документом ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень» наступні: у теплий період року температура повітря – 23-25 градусів, відносна вологість – 60-40, швидкість руху – 0,1 м/сек.; у холодний період року температура повітря – 22-24 градусів, відносна вологість – 60-40, швидкість руху

– 0,1 м/сек.

Забезпечення пожежної безпеки є обов'язковою складовою, основними напрямками якої є ліквідування умов виникнення пожежі та зведення до мінімуму її наслідків. По-перше потрібно визначити категорію пожежної небезпеки. Площа приміщення, яке розглядається – 17 кв. м.. Вибір типу та необхідної кількості вогнегасників проводиться згідно з нормами належності. Виходячи з НАПБ Б.03.002-2007 «Норми визначення категорій приміщень, будівель та зовнішніх установок щодо вибухонебезпечної та пожежної небезпеки» [12] можна визначити категорію даного приміщення. Є п'ять категорій, приміщення належить до категорії В, так як у ньому не знаходяться горючі гази, пил або волокна, легкозаймисті рідини у тій кількості та з такою температурою, яка описана у таблиці «Категорії приміщень і будівель за вибуховою та пожежною небезпекою» документа.

Причини виникнення пожежі у даному приміщенні можуть бути наступні:

- випадкове потрапляння води на електроприлади;
- порушення певних правил пожежної безпеки при влаштуванні або експлуатації електрообладнання;
- коротке замикання;
- перевантаження електромережі;
- неправильні дії під час займання, що може призвести до швидкого поширення полум'я;
- паління в приміщенні.

Первинними засобами пожежного захисту є наступне:

- вогнегасники;
- пожежний інвентар;
- пожежний інструмент.

Серед усіх первинних засобів пожежогасіння, які використовуються, найбільш ефективним віддається роль вогнегасникам. Встановлено, що з використанням вогнегасників можна ліквідувати пожежу навіть протягом перших

чотирьох хвилин з моменту її виникнення. Вогнегасники поділяються за способом транспортування на переносні та пересувні, за видом вогнегасної речовини на водопінний, водяний, водопінний аерозольний, вуглекислотний і порошковий, адже для кожного способу припинення горіння відповідає конкретний вид вогнегасної речовини.

У даному випадку, тобто аналізуючи приміщення, можна зробити висновок що повинен бути вуглекислотний вогнегасник на 2 кг, але він для тих випадків, коли трапилось легке займання електрообладнання, так як даний вогнегасник у більш легкій формі впливає на електричне обладнання, а якщо відбулося більш масштабне займання, тобто і стіл і шафа теж почали горіти, в таких випадках повинен бути порошковий вогнегасник теж на 2 кг, потрібно пам'ятати що після такого вогнегасника експлуатація електрообладнання вже буде неможливою.

Також до засобів пожежного захисту можна віднести:

- системи пожежної сигналізації та зв'язку (для того щоб досить швидко виявити, сигналізувати, оповістити людей, які знаходяться у даному приміщенні про виникнення пожежі та своєчасно викликати пожежні підрозділи використовують дані системи; більш швидкою та надійною вважається автоматична установка пожежної сигналізації, яка може працювати цілодобово);
- системи пожежогасіння (дана система є однією з найефективніших засобів пожежного захисту, частіше за все вона використовується разом з системою пожежної сигналізації та зв'язку, яка описана вище, існують такі види розглядаємої системи: спринклерні, дренчерні, газові, прошкові, аерозольні та системи пожежогасіння з тонко розпиленою водою, основною її функцією є безпосередньо пожежогасіння після оповіщення про виникнення пожежі);
- план евакуації (обов'язково повинен бути даний засіб пожежного захисту у приміщенні, адже потрібно позначити певні шляхи і евакуаційні виходи за якими у випадку виникнення пожежі буде відбуватись самостійний вихід людей з приміщення).

Отже, незалежно від виду професійної діяльності питання охорони праці завжди є актуальним. Дуже важливо правильно оцінити небезпечні та шкідливі виробничі фактори, адже від цього залежить забезпечення здорових та безпечних умов праці, також обов'язково потрібно приділити вагому увагу пожежній безпеці.

Аналізуючи даний об'єкт на предмет безпеки було виявлено небезпечний виробничий фактор: небезпека ураження електричним струмом; шкідливі виробничі фактори: недостатня освітленість робочої зони, підвищений рівень шуму, вимушена робоча поза, дрібні стереотипні рухи пальцями рук, розумові напруження, емоційне напруження та був проведений аналіз параметрів мікроклімату і пожежної безпеки, надані певні засоби захисту та рекомендації для кожного пункту.

## ВИСНОВКИ

У ході виконання даної кваліфікаційної роботи, мета, яка ставилась, а саме розробка системи для планування навчальної роботи кафедри шляхом автоматизації та формування навчального плану освітньої програми була досягнутою.

Було проведено огляд існуючих баз даних та програм для автоматизації вищих навчальних закладів. У ході аналізу виявилось що нема такої програми, яка б не тільки містила у собі базу даних кафедри, а й у якій можна було б створювати навчальні та робочі навчальні плани, також у якій була б можливість вносити певні корективи, виходячи з нових норм, перевіряючи користувача у ході роботи та яка була б зручною для використання.

Був проведений аналіз предметної області та визначено структуру системи, яка розробляється. Був запропонований певний перелік довідників та документів, які будуть складати систему, яка розробляється. Також була спроектована система планування навчальної роботи кафедри. У роботі наводиться опис спроектованої системи у вигляді ERD діаграми.

Розроблена система була програмно реалізована у вигляді CRM-системи, вона зможе полегшити роботу з досить відповідальними та складними документами, а точніше з навчальними та робочими навчальними планами та допоможе з необхідністю поновлення, виходячи з нових норм і тим самим зможе підвищити якість навчального процесу.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Що таке база даних, їх призначення та види. URL: <https://futurenow.com.ua/shho-take-bazy-danyh-yih-pryznachennya-ta-vydy/>
2. Основні відомості про бази даних. URL: <https://support.microsoft.com/uk-ua/office>
3. Бодненко Д.М., Кулик Є.С., Алексеєва К.М., Гулієва Н.С., П'янкова Є.І., Хорець І.В. Використання хмарних сервісів у освітньому процесі: google keep. Київ: Київський університет імені Бориса Грінченка, 2021
4. phpMyAdmin *Матеріал з вікіпедії – вільної енциклопедії* URL: <https://uk.wikipedia.org/wiki/PhpMyAdmin>
5. Система управління базами даних (СУБД). СУБД Microsoft Access. URL: <https://sites.google.com/site/tehnikakomp/home/samostijne-vivcenna-materialu/sistema-upravlinna-bazami-danih-subd-subd-microsoft-access>
6. aSc TimeTables - JSC International Business Network. URL: <http://www.ibn.lt/asc-timetables/?lang=en>
7. Використання Microsoft Excel (зведена таблиця): Переваги/ Переваги та недоліки. URL: <https://www.alltechbuzz.net/uk/15-reasons-why-you-should-use-microsoft-excel/>
8. Лосєв М.Ю. Бази Даних. Харків: ХНЕУ ім. С. Кузнеця, 2018.
9. Що таке CRM-система: повний гід по вибору CRM для початківців. URL: <https://nethunt.ua/blog/shcho-takie-crm-sistiema-povnii-ghid-po-viboru-crm-dlia-pochatktivsiv-v-2020/>
10. Переваги 1С Підприємство – основні переваги та недоліки програми. URL: <https://rentsoft.ua/ua/blog/preimushchestva-1c-predpriyatie-osnovnye-dostoinstva-i-nedostatki-programmy/index.php>
11. ДСН 3.3.6.037-99 Санітарні норми виробничого шуму, ультразвуку та інфразвуку.
12. НАПБ Б.03.002-2007 Норми визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою.



## Додаток А. Лістинг програми

```

//код програми для форми документа навчального плану
&НаКлиенте
Процедура ПрограмаПриИзменении(Элемент)
    Объект.Специальность = Специальность(Объект.Программа);
КонецПроцедуры
&НаСервере
Функция Специальность(ВыбПрограмма)
    Возврат ОбщегоНазначения.ПолучитьСпециальность(ВыбПрограмма);
КонецФункции
&НаКлиенте
Процедура УчебныеДисциплиныДисциплинаПриИзменении(Элемент)
    ТекущиеДанные = Элементы.УчебныеДисциплины.ТекущиеДанные;
    ТекущиеДанные.Шифр = Шифр(ТекущиеДанные.Дисциплина);
КонецПроцедуры
&НаСервере
Функция Шифр(ВыбДисциплина)
    Возврат ОбщегоНазначения.ПолучитьШифрДисциплины(ВыбДисциплина);
КонецФункции
&НаКлиенте
Процедура ЗаполнитьДисциплинами(Команда)
    список = Новый СписокЗначений(); //список.Добавить(
    список.Добавить("ЗаполнитьВсе", "Все дисциплины",);
    список.Добавить("ЗаполнитьОбязательные", "Обязательные
дисциплины",);
    список.Добавить("ЗаполнитьВыборочные", "Выборочные
дисциплины",);
    ВыбЭлемент = список.ВыбратьЭлемент();
    Если ВыбЭлемент = Неопределено Тогда
        Возврат;
    Иначе
        Если ВыбЭлемент.Значение = "ЗаполнитьВсе" Тогда
            ЗаполнитьВсеДисциплины();
        ИначеЕсли ВыбЭлемент.Значение = "ЗаполнитьОбязательные"
Тогда
            ЗаполнитьОбязательныеВсе();
        ИначеЕсли ВыбЭлемент.Значение = "ЗаполнитьВыборочные" Тогда
            ЗаполнитьВыборочныеВсе();
        КонецЕсли;
    КонецЕсли;
КонецПроцедуры
&НаКлиенте
Процедура ЗаполнитьВсеДисциплины()
    список = Новый СписокЗначений();
    СформироватьЗапросПоДисциплинам(список, "Все");

    Объект.Обязательные.Очистить();
    Объект.Выборочные.Очистить();
    ЗаполнитьТабличнуюЧастьОбязательные(список);
    ЗаполнитьТабличнуюЧастьВыборочные(список);

```

```

КонецПроцедуры
&НаКлиенте
Процедура ЗаполнитьОбязательныеВсе ()
    спис = Новый СписокЗначений ();
    СформироватьЗапросПоДисциплинам (спис, "Обязательные");

    Объект.Обязательные.Очистить ();
    ЗаполнитьТабличнуюЧастьОбязательные (спис);

    Для Каждого ТекСтрока Из Объект.Обязательные Цикл
        ТекстСообщения =
ПроверитьКредитыИОбщийОбъем (ТекСтрока.Дисциплина, ТекСтрока.Кредиты,
ТекСтрока.ОбщийОбъем, ТекСтрока.СРС, ТекСтрока.ВсегоАудиторные);
        Если ЗначениеЗаполнено (ТекстСообщения) Тогда
            Сообщить (" "+ТекстСообщения);
        КонецЕсли;
    КонецЦикла;
КонецПроцедуры
&НаКлиенте
Процедура ЗаполнитьВыборочныеВсе ()
    спис = Новый СписокЗначений ();
    СформироватьЗапросПоДисциплинам (спис, "Выборочные");
    Объект.Выборочные.Очистить ();
    ЗаполнитьТабличнуюЧастьВыборочные (спис);
    Для Каждого ТекСтрока Из Объект.Выборочные Цикл
        ТекстСообщения =
ПроверитьКредитыИОбщийОбъем (ТекСтрока.Дисциплина, ТекСтрока.Кредиты,
ТекСтрока.ОбщийОбъем, ТекСтрока.СРС, ТекСтрока.ВсегоАудиторные);
        Если ЗначениеЗаполнено (ТекстСообщения) Тогда
            Сообщить (" "+ТекстСообщения);
        КонецЕсли;
    КонецЦикла;
КонецПроцедуры
&НаСервере
Процедура СформироватьЗапросПоДисциплинам (спис, видДисциплины)
    Запрос = Новый Запрос;
    Запрос.Текст =
        "ВЫБРАТЬ
        |     Дисциплина.Ссылка КАК Ссылка,
        |     Дисциплина.Специальность КАК Специальность,
        |     Дисциплина.Программа КАК Программа,
        |     Дисциплина.Уровень КАК Уровень
        |ИЗ
        |     Справочник.Дисциплина КАК Дисциплина
        |ГДЕ
        |     Дисциплина.Специальность = &Специальность
        |     И Дисциплина.Программа = &Программа
        |     И Дисциплина.Уровень = &Уровень";

    Запрос.УстановитьПараметр ("Программа", Объект.Программа);
    Запрос.УстановитьПараметр ("Специальность",
Объект.Специальность);
    Запрос.УстановитьПараметр ("Уровень", Объект.Уровень);

```

```

    РезультатЗапроса = Запрос.Выполнить ();
    ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать ();
    Пока ВыборкаДетальныеЗаписи.Следующий () Цикл
        поиск =
ВыборкаДетальныеЗаписи.Ссылка.Атуальность.Выгрузить ();
        НайденнаяСтрока = поиск.Найти (Объект.УчебныйГод,
"УчебныйГод" );
        Если НайденнаяСтрока = Неопределено Тогда
            Продолжить;
        КонецЕсли;
        Если видДисциплины = "Все" Тогда
            спис.Добавить (ВыборкаДетальныеЗаписи.Ссылка);
        ИначеЕсли видДисциплины = "Обязательные" Тогда
            Если ВыборкаДетальныеЗаписи.Ссылка.Часть =
Справочники.ЧастиПлана.Обязательная Тогда
                спис.Добавить (ВыборкаДетальныеЗаписи.Ссылка);
            КонецЕсли;
            ИначеЕсли видДисциплины = "Выборочные" Тогда
                Если ВыборкаДетальныеЗаписи.Ссылка.Часть =
Справочники.ЧастиПлана.Выборочная Тогда
                    спис.Добавить (ВыборкаДетальныеЗаписи.Ссылка);
                КонецЕсли;
            КонецЕсли;
        КонецЦикла;
    КонецПроцедуры
&НаСервере
Процедура СоздатьТаблицуЗначенийСоставДисциплины (тз)
    тз.Колонки.Добавить ("НомСем" );
    тз.Колонки.Добавить ("Контроль" );
    тз.Колонки.Добавить ("Работы" );
    тз.Колонки.Добавить ("Кредиты" );
    тз.Колонки.Добавить ("ОбщийОбъем" );
    тз.Колонки.Добавить ("Лекций" );
    тз.Колонки.Добавить ("Практик" );
    тз.Колонки.Добавить ("Лаб" );
    тз.Колонки.Добавить ("СРС" );
КонецПроцедуры
&НаСервере
Процедура ЗаполнитьТабличнуюЧастьОбязательные (спис)
    тз = Новый ТаблицаЗначений;
    СоздатьТаблицуЗначенийСоставДисциплины (тз);
    Для к = 0 По спис.Количество() - 1 Цикл
        Дисцип = спис.Получить (к).Значение;
        Если Дисцип.Часть = Справочники.ЧастиПлана.Обязательная
Тогда
            тз.Очистить ();
            ЗаполнитьИзСоставаДисциплины (Дисцип, тз);
            Если тз.Количество() = 0 Тогда
                НовСтрока = Объект.Обязательные.Добавить ();
                НовСтрока.Дисциплина = Дисцип;
                НовСтрока.Шифр = НовСтрока.Дисциплина.Шифр;
            Иначе
                Для Каждого ТекСтрока Из тз Цикл

```

```

НовСтрока = Объект.Обязательные.Добавить ();
НовСтрока.Дисциплина = Дисцип;
НовСтрока.Шифр = НовСтрока.Дисциплина.Шифр;
НовСтрока.НомерСеместра = ТекСтрока.НомСем;
НовСтрока.СеместровыйКонтроль =
ТекСтрока.Контроль;
НовСтрока.СеместровыеРаботы =
ТекСтрока.Работы;
НовСтрока.Кредиты = ТекСтрока.Кредиты;
НовСтрока.ОбщийОбъем = ТекСтрока.ОбщийОбъем;
НовСтрока.Лекций = ТекСтрока.Лекций;
НовСтрока.Практик = ТекСтрока.Практик;
НовСтрока.Лабораторных = ТекСтрока.Лаб;
НовСтрока.СРС = ТекСтрока.СРС;
НовСтрока.ВсегоАудиторные = НовСтрока.Лекций
+ НовСтрока.Практик + НовСтрока.Лабораторных;
    КонечЦикла;
        КонечЕсли;
            КонечЦикла;
                Объект.Обязательные.Сортировать ("НомерСеместра");
КонечПроцедуры
&НаСервере
Процедура ЗаполнитьТабличнуюЧастьВыборочные (спис)
    тз = Новый ТаблицаЗначений;
    СоздатьТаблицуЗначенийСоставДисциплины(тз);
    Для к = 0 По спис.Количество() - 1 Цикл
        Дисцип = спис.Получить(к).Значение;
        тз.Очистить();
        ЗаполнитьИзСоставаДисциплины(Дисцип, тз);
        Если Дисцип.Часть = Справочники.ЧастиПлана.Выборочная Тогда
            тз.Очистить();
            ЗаполнитьИзСоставаДисциплины(Дисцип, тз);
            Если тз.Количество() = 0 Тогда
                НовСтрока = Объект.Выборочные.Добавить();
                НовСтрока.Дисциплина = Дисцип;
                НовСтрока.Шифр = НовСтрока.Дисциплина.Шифр;
            Иначе
                Для Каждого ТекСтрока Из тз Цикл
                    НовСтрока = Объект.Выборочные.Добавить();
                    НовСтрока.Дисциплина = Дисцип;
                    НовСтрока.Шифр = НовСтрока.Дисциплина.Шифр;
                    НовСтрока.НомерСеместра = ТекСтрока.НомСем;
                    НовСтрока.СеместровыйКонтроль =
ТекСтрока.Контроль;
                    НовСтрока.СеместровыеРаботы =
ТекСтрока.Работы;
                    НовСтрока.Кредиты = ТекСтрока.Кредиты;
                    НовСтрока.ОбщийОбъем = ТекСтрока.ОбщийОбъем;
                    НовСтрока.Лекций = ТекСтрока.Лекций;
                    НовСтрока.Практик = ТекСтрока.Практик;
                    НовСтрока.Лабораторных = ТекСтрока.Лаб;
                    НовСтрока.СРС = ТекСтрока.СРС;

```

```

НовСтрока.ВсегоАудиторные = НовСтрока.Лекций
+ НовСтрока.Практик + НовСтрока.Лабораторных;
    КонечЦикла;
        КонечЕсли;
            КонечЕсли;
                КонечЦикла;
                Объект.Выборочные.Сортировать ("Шифр");
КонечПроцедуры
&НаСервере
Процедура ЗаполнитьИзСоставаДисциплины(Дисцип, тз)
    Запрос = Новый Запрос;
    Запрос.Текст =
        "ВЫБРАТЬ
        | СоставДисциплиныСрезПоследних.Дисциплина КАК
Дисциплина,
        | СоставДисциплиныСрезПоследних.НомерСеместра КАК
НомерСеместра,
        | СоставДисциплиныСрезПоследних.СеместровыйКонтроль КАК
СеместровыйКонтроль,
        | СоставДисциплиныСрезПоследних.СеместровыеРаботы КАК
СеместровыеРаботы,
        | СоставДисциплиныСрезПоследних.Кредиты КАК Кредиты,
        | СоставДисциплиныСрезПоследних.ОбщийОбъем КАК
ОбщийОбъем,
        | СоставДисциплиныСрезПоследних.Лекций КАК Лекций,
        | СоставДисциплиныСрезПоследних.Практик КАК Практик,
        | СоставДисциплиныСрезПоследних.Лабораторных КАК
Лабораторных,
        | СоставДисциплиныСрезПоследних.СРС КАК СРС,
        | СоставДисциплиныСрезПоследних.ФормаОбучения КАК
ФормаОбучения
        | ИЗ
        |
        РегистрСведений.СоставДисциплины.СрезПоследних (&ВыбДата,
Дисциплина = &ВыбДисц) КАК СоставДисциплиныСрезПоследних
        | ГДЕ
        | СоставДисциплиныСрезПоследних.ФормаОбучения =
&ФормаОбучения";
    Запрос.УстановитьПараметр ("ВыбДата", Объект.Дата);
    Запрос.УстановитьПараметр ("ВыбДисц", Дисцип);
    Запрос.УстановитьПараметр ("ФормаОбучения",
Объект.ФормаОбучения);
    РезультатЗапроса = Запрос.Выполнить();
    ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();
    Пока ВыборкаДетальныеЗаписи.Следующий() Цикл
        НовСтрока = тз.Добавить();
        НовСтрока.НомСем = ВыборкаДетальныеЗаписи.НомерСеместра;
        НовСтрока.Контроль =
ВыборкаДетальныеЗаписи.СеместровыйКонтроль;
        НовСтрока.Работы =
ВыборкаДетальныеЗаписи.СеместровыеРаботы;
        НовСтрока.Кредиты = ВыборкаДетальныеЗаписи.Кредиты;
        НовСтрока.ОбщийОбъем = ВыборкаДетальныеЗаписи.ОбщийОбъем;

```

```

НовСтрока.Лекций = ВыборкаДетальныеЗаписи.Лекций;
НовСтрока.Практик = ВыборкаДетальныеЗаписи.Практик;
НовСтрока.Лаб = ВыборкаДетальныеЗаписи.Лабораторных;
НовСтрока.СРС = ВыборкаДетальныеЗаписи.СРС;

```

```

КонецЦикла;

```

```

КонецПроцедуры

```

```

&НаКлиенте

```

```

Процедура УчебныеДисциплиныПриОкончанииРедактирования (Элемент,
НоваяСтрока, ОтменаРедактирования)

```

```

    ТекущиеДанные = Элементы.УчебныеДисциплины.ТекущиеДанные;

```

```

    ТекущиеДанные.ВсегоАудиторные = ТекущиеДанные.Лекций +

```

```

    ТекущиеДанные.Практик + ТекущиеДанные.Лабораторных;

```

```

    ТекущиеДанные.ОбщийОбъем = ТекущиеДанные.Лекций +

```

```

    ТекущиеДанные.Практик + ТекущиеДанные.Лабораторных +

```

```

    ТекущиеДанные.СРС;

```

```

КонецПроцедуры

```

```

&НаКлиенте

```

```

Процедура ОбязательныеПриОкончанииРедактирования (Элемент,
НоваяСтрока, ОтменаРедактирования)

```

```

    ТекущиеДанные = Элементы.Обязательные.ТекущиеДанные;

```

```

    ТекущиеДанные.ВсегоАудиторные = ТекущиеДанные.Лекций +

```

```

    ТекущиеДанные.Практик + ТекущиеДанные.Лабораторных;

```

```

    ТекущиеДанные.ОбщийОбъем = ТекущиеДанные.Лекций +

```

```

    ТекущиеДанные.Практик + ТекущиеДанные.Лабораторных +

```

```

    ТекущиеДанные.СРС;

```

```

    ТекстСообщения =

```

```

    ПроверитьКредитыИОбщийОбъем (ТекущиеДанные.Дисциплина,

```

```

    ТекущиеДанные.Кредиты, ТекущиеДанные.ОбщийОбъем, ТекущиеДанные.СРС,

```

```

    ТекущиеДанные.ВсегоАудиторные);

```

```

    Если ЗначениеЗаполнено (ТекстСообщения) Тогда

```

```

        Сообщить (" "+ТекстСообщения);

```

```

    КонецЕсли;

```

```

КонецПроцедуры

```

```

&НаКлиенте

```

```

Процедура ВыборочныеПриОкончанииРедактирования (Элемент, НоваяСтрока,
ОтменаРедактирования)

```

```

    ТекущиеДанные = Элементы.Выборочные.ТекущиеДанные;

```

```

    ТекущиеДанные.ВсегоАудиторные = ТекущиеДанные.Лекций +

```

```

    ТекущиеДанные.Практик + ТекущиеДанные.Лабораторных;

```

```

    ТекущиеДанные.ОбщийОбъем = ТекущиеДанные.Лекций +

```

```

    ТекущиеДанные.Практик + ТекущиеДанные.Лабораторных +

```

```

    ТекущиеДанные.СРС;

```

```

    ТекстСообщения =

```

```

    ПроверитьКредитыИОбщийОбъем (ТекущиеДанные.Дисциплина,

```

```

    ТекущиеДанные.Кредиты, ТекущиеДанные.ОбщийОбъем, ТекущиеДанные.СРС,

```

```

    ТекущиеДанные.ВсегоАудиторные);

```

```

    Если ЗначениеЗаполнено (ТекстСообщения) Тогда

```

```

        Сообщить (" "+ТекстСообщения);

```

```

    КонецЕсли;

```

```

КонецПроцедуры

```

```

&НаКлиенте

```

```

Процедура ОбязательныеДисциплинаПриИзменении (Элемент)

```

```

    ТекущиеДанные = Элементы.Обязательные.ТекущиеДанные;

```

```

ТекущиеДанные.Шифр = Шифр (ТекущиеДанные.Дисциплина);
КонецПроцедуры
&НаСервере
Функция ПроверитьКредитыИОбщийОбъем (ВыбДисциплина, ВыбКредиты,
ВыбОбщийОбъем, ВыбСРС, ВыбАудит)
    ТекстСообщения = "";
    ТекНастройка = "";
    НайденнаяСсылка =
Справочники.Настройки.НайтиПоРеквизиту ("УчебныйГод", Объект.УчебныйГо
д);
    Если НайденнаяСсылка <> Справочники.Настройки.ПустаяСсылка ()
Тогда
        ТекНастройка = НайденнаяСсылка;
    КонецЕсли;
    Если НЕ ЗначениеЗаполнено (ТекНастройка) Тогда
        ТекстСообщения = "Нет данных про количество часов в 1
кредите для "+Объект.УчебныйГод.Наименование;
        Возврат ТекстСообщения;
    КонецЕсли;
    Если ВыбОбщийОбъем / ТекНастройка.КоличествоЧасовВКредите <>
ВыбКредиты Тогда
        ТекстСообщения = "У дисциплины " + ВыбДисциплина + " не
правильное соотношение кредиты - общий объем" + Символы.ПС;
        Врем = ВыбСРС;
        ВыбСРС = ВыбСРС + (ВыбКредиты *
ТекНастройка.КоличествоЧасовВКредите - ВыбОбщийОбъем);
        ВыбОбщийОбъем = ВыбАудит + ВыбСРС;
        ТекстСообщения = ТекстСообщения + "Произведен перерасчет
СРС с "+ Врем + " на " + ВыбСРС + Символы.ПС;
    КонецЕсли;
    Возврат ТекстСообщения;
КонецФункции
&НаКлиенте
Процедура ВыборочныеДисциплинаПриИзменении (Элемент)
    ТекущиеДанные = Элементы.Выборочные.ТекущиеДанные;
    ТекущиеДанные.Шифр = Шифр (ТекущиеДанные.Дисциплина);
КонецПроцедуры
&НаКлиенте
Процедура ФормаОбученияПриИзменении (Элемент)
    УстВидимость ();
КонецПроцедуры
&НаСервере
Процедура ПриСозданииНаСервере (Отказ, СтандартнаяОбработка)
    Элементы.ОбязательныеКонтрольныеРаботы.Видимость = Ложь;
КонецПроцедуры
&НаСервере
Процедура УстВидимость ()
    Если Объект.ФормаОбучения =
Перечисления.ВидыФормОбучения.Заочная Тогда
        Элементы.ОбязательныеКонтрольныеРаботы.Видимость = Истина;
    Иначе
        Элементы.ОбязательныеКонтрольныеРаботы.Видимость = Ложь;
    КонецЕсли;

```

```

КонецПроцедуры
&НаКлиенте
Процедура ПриОткрытии (Отказ)
    УстВидимость ();
КонецПроцедуры
&НаКлиенте
Процедура ЗаполнитьДисциплинамиПодбор (Команда)
КонецПроцедуры
&НаКлиенте
Процедура ЗаполнитьСоставДляВсех (Команда)
    ПолучитьСоставДисциплин ();
КонецПроцедуры
&НаСервере
Процедура ПолучитьСоставДисциплин ()
    Запрос = Новый Запрос;
    Запрос.Текст =
        "ВЫБРАТЬ
            | СоставДисциплиныСрезПоследних.УчебныйГод КАК
УчебныйГод,
            | СоставДисциплиныСрезПоследних.Уровень КАК Уровень,
            | СоставДисциплиныСрезПоследних.Специальность КАК
Специальность,
            | СоставДисциплиныСрезПоследних.Программа КАК Программа,
            | СоставДисциплиныСрезПоследних.Квалификация КАК
Квалификация,
            | СоставДисциплиныСрезПоследних.ФормаОбучения КАК
ФормаОбучения,
            | СоставДисциплиныСрезПоследних.Дисциплина КАК
Дисциплина,
            | СоставДисциплиныСрезПоследних.НомерСеместра КАК
НомерСеместра,
            | СоставДисциплиныСрезПоследних.СеместровыйКонтроль КАК
СеместровыйКонтроль,
            | СоставДисциплиныСрезПоследних.СеместровыеРаботы КАК
СеместровыеРаботы,
            | СоставДисциплиныСрезПоследних.Кредиты КАК Кредиты,
            | СоставДисциплиныСрезПоследних.ОбщийОбъем КАК
ОбщийОбъем,
            | СоставДисциплиныСрезПоследних.Лекций КАК Лекций,
            | СоставДисциплиныСрезПоследних.Практик КАК Практик,
            | СоставДисциплиныСрезПоследних.Лабораторных КАК
Лабораторных,
            | СоставДисциплиныСрезПоследних.СРС КАК СРС
            | ИЗ
            | РегистрСведений.СоставДисциплины.СрезПоследних (&Дата1,
) КАК СоставДисциплиныСрезПоследних
            | ГДЕ
            | СоставДисциплиныСрезПоследних.Уровень = &Уровень
            | И СоставДисциплиныСрезПоследних.Специальность =
&Специальность
            | И СоставДисциплиныСрезПоследних.Программа = &Программа
            | И СоставДисциплиныСрезПоследних.Квалификация =
&Квалификация

```



```

|      И СоставДисциплиныСрезПоследних.ФормаОбучения =
&ФормаОбучения
|      И СоставДисциплиныСрезПоследних.УчебныйГод =
&УчебныйГод
|      И СоставДисциплиныСрезПоследних.Дисциплина В(&спис)";
Запрос.УстановитьПараметр("Дата1", Объект.Дата);
Запрос.УстановитьПараметр("Квалификация", Объект.Квалификация);
Запрос.УстановитьПараметр("Программа", Объект.Программа);
Запрос.УстановитьПараметр("Специальность",
Объект.Специальность);
Запрос.УстановитьПараметр("Уровень", Объект.Уровень);
Запрос.УстановитьПараметр("УчебныйГод", Объект.УчебныйГод);
Запрос.УстановитьПараметр("ФормаОбучения",
Объект.ФормаОбучения);
РезультатЗапроса = Запрос.Выполнить();
ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();
Пока ВыборкаДетальныеЗаписи.Следующий() Цикл
КонецЦикла;
КонецПроцедуры
&НаКлиенте
Процедура Проверить(Команда)
ТекстСообщения = "";
ПараметрыФормы = Новый Структура("Документ, Сообщение",
Объект.Ссылка, ТекстСообщения);
ОткрытьФорму("Документ.УчебныйПлан.Форма.ФормаПроверка",
ПараметрыФормы);
КонецПроцедуры

//код програми для форми перевірки навчального плану
&НаСервере
Процедура ПриСозданииНаСервере(Отказ, СтандартнаяОбработка)
Докум = Параметры.Документ;
Сообщение = Параметры.Сообщение;
КонецПроцедуры
&НаКлиенте
Процедура ПриОткрытии(Отказ)
ПроверитьСоответствияНастройкам()
КонецПроцедуры
&НаСервере
Функция ПроверитьСоответствияНастройкам()
ТекстСообщения = "";
ТекНастройка = "";
НайденнаяСсылка =
Справочники.Настройки.НайтиПоРеквизиту("УчебныйГод", Докум.УчебныйГод
);
Если НайденнаяСсылка <> Справочники.Настройки.ПустаяСсылка()
Тогда
ТекНастройка = НайденнаяСсылка;
КонецЕсли;

Если НЕ ЗначениеЗаполнено(ТекНастройка) Тогда
ТекстСообщения = "Нет данных для
"+Докум.УчебныйГод.Наименование;

```

```

        Возврат ТекстСообщения;
    КонецЕсли;
    ТекстСообщения =
ПроверитьСоответствиеАудиторнойНагрузке (ТекНастройка) ;
    Возврат ТекстСообщения;
КонецФункции
&НаСервере
Функция ПроверитьСоответствиеАудиторнойНагрузке (ТекНастройка)
    ТекстСообщения = "";
    врем = Докум.Обязательные.Выгрузить ();
    врем.Очистить ();
    Для Каждого ТекСтрока из Докум.Обязательные Цикл
        Если (ТекСтрока.Дисциплина.Внекредитная) или
(ТекСтрока.Дисциплина.Практика) или
(ТекСтрока.Дисциплина.КвалификационнаяРабота) Тогда
            Продолжить;
        КонецЕсли;
        СтрокаТЗ = врем.Добавить ();
        ЗаполнитьЗначенияСвойств (СтрокаТЗ, ТекСтрока);
    КонецЦикла;
    Для Каждого ТекСтрока из Докум.Выборочные Цикл
        Если (ТекСтрока.Дисциплина.Внекредитная) или
(ТекСтрока.Дисциплина.Практика) или
(ТекСтрока.Дисциплина.КвалификационнаяРабота) Тогда
            Продолжить;
        КонецЕсли;
        СтрокаТЗ = врем.Добавить ();
        ЗаполнитьЗначенияСвойств (СтрокаТЗ, ТекСтрока);
    КонецЦикла;
    Для Каждого ТекСтрока из врем Цикл
        СтрокаТЗ = тз.Добавить ();
        ЗаполнитьЗначенияСвойств (СтрокаТЗ, ТекСтрока);
        Если СтрокаТЗ.Дисциплина.Часть =
Справочники.ЧастиПлана.Выборочная Тогда
            СтрокаТЗ.Выборочная = Истина;
            СтрокаТЗ.Кво = 1;
        КонецЕсли;
    КонецЦикла;
    Для Каждого ТекСтрока из тз Цикл
        СтрокаТЗ = тз1.Добавить ();
        ЗаполнитьЗначенияСвойств (СтрокаТЗ, ТекСтрока);
    КонецЦикла;
    тз1.Сортировать ("НомерСеместра");
    тз3 = тз1.Выгрузить ();
    тз3.Свернуть ("НомерСеместра,Выборочная", "Кво, Кредиты,
ВсегоАудиторные");
    тз2 = тз1.Выгрузить ();
    тз2.Свернуть ("НомерСеместра", "Кредиты, ВсегоАудиторные");
    ТекстСообщения = "№ семестра Кредиты Аудит."+ Символы.ПС;
    ТекстСообщения = ТекстСообщения + "-----"
"+ Символы.ПС;
    Если Докум.Уровень =
Перечисления.УровниВысшегоОбразования.Магистр Тогда

```

```

        Для Каждого ТекСтрока Из ТекНастройка.НастройкаМагистры
Цикл
        ТекстСообщения = ТекстСообщения + ТекСтрока.Семестр+
": " + "Норма аудиторной - " + (ТекСтрока.МаксАудиторная *
ТекСтрока.КвоНедель) + Символы.ПС;
        квоВыб = 0; квоНадо = ТекСтрока.КвоВыборочных;
        квоКредиты = 0; КвоАудит = 0; квоКредитыВыб = 0;
КвоАудитВыб = 0;
        Для Каждого тСтрока1 Из тз3 Цикл
            Если (тСтрока1.НомерСеместра = ТекСтрока.Семестр)
и (тСтрока1.Выборочная = Истина) Тогда
                квоВыб = тСтрока1.Кво;
                КвоАудитВыб = КвоАудитВыб +
тСтрока1.ВсегоАудиторные;
                КвоКредитыВыб = КвоКредитыВыб +
тСтрока1.Кредиты;
            КонечЕсли;
            Если (тСтрока1.НомерСеместра = ТекСтрока.Семестр)
и (тСтрока1.Выборочная = Ложь) Тогда
                КвоАудит = КвоАудит +
тСтрока1.ВсегоАудиторные;
                КвоКредиты = квоКредиты + тСтрока1.Кредиты;
            КонечЕсли;
        КонечЦикла;
        Если квоВыб = 0 Тогда
            ТекстСообщения = ТекстСообщения +
ТекСтрока.Семестр+ " - " + квоКредиты + " - " + КвоАудит +
Символы.ПС;
        Иначе
            пересчетКредиты = (КвоКредитыВыб / квоВыб) *
квоНадо;
            пересчетАудит = (КвоАудитВыб / квоВыб) * квоНадо;
            ТекстСообщения = ТекстСообщения +
ТекСтрока.Семестр + " - " + (квоКредиты+пересчетКредиты) + " - " +
(КвоАудит+пересчетАудит) + Символы.ПС;
        КонечЕсли;
    КонечЦикла;
Иначе
    КонечЕсли;
        Сообщение = ТекстСообщения;
        Возврат ТекстСообщения;
КонечФункции

//код програми для команды друку навчального плану
&НаКлиенте
Процедура ОбработкаКоманды(ПараметрКоманды,
ПараметрыВыполненияКоманды)
    ТабДок = Новый ТабличныйДокумент;
    Печать(ТабДок, ПараметрКоманды);
    ТабДок.АвтоМасштаб = Истина;
    ТабДок.ОриентацияСтраницы = ОриентацияСтраницы.Ландшафт;
    ТабДок.Показать(ПараметрКоманды);
КонечПроцедуры

```

```

&НаСервере
Процедура Печать (ТабДок, СсылкаНаДокумент)
    Если СсылкаНаДокумент.Уровень =
Перечисления.УровниВысшегоОбразования.Бакалавр Тогда
    Макет =
Документы.УчебныйПлан.ПолучитьМакет ("МакетБакалавры_2021");
    КонецЕсли;
    Запрос = Новый Запрос;
    Запрос.Текст =
"ВЫБРАТЬ
|     УчебныйПлан.УчебныйГод КАК УчебныйГод,
|     УчебныйПлан.Уровень КАК Уровень,
|     УчебныйПлан.Специальность КАК Специальность,
|     УчебныйПлан.Программа КАК Программа,
|     УчебныйПлан.Квалификация КАК Квалификация,
|     УчебныйПлан.ФормаОбучения КАК ФормаОбучения
| ИЗ
|     Документ.УчебныйПлан КАК УчебныйПлан
| ГДЕ
|     УчебныйПлан.Ссылка = &СсылкаНаДокумент";

    Запрос.УстановитьПараметр ("СсылкаНаДокумент",
СсылкаНаДокумент);
    РезультатЗапроса = Запрос.Выполнить ();
    ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать ();
    ОбластьШапка = Макет.ПолучитьОбласть ("Шапка");
    Пока ВыборкаДетальныеЗаписи.Следующий () Цикл
        //ОбластьШапка.Параметры.УчебныйГод =
ВыборкаДетальныеЗаписи.УчебныйГод;
        ОбластьШапка.Параметры.Уровень =
ВыборкаДетальныеЗаписи.Уровень;
        ОбластьШапка.Параметры.Специальность =
ВыборкаДетальныеЗаписи.Специальность;
        ОбластьШапка.Параметры.Программа =
ВыборкаДетальныеЗаписи.Программа;
        ОбластьШапка.Параметры.Квалификация =
ВыборкаДетальныеЗаписи.Квалификация;
        ОбластьШапка.Параметры.ФормаОбучения =
ВыборкаДетальныеЗаписи.ФормаОбучения;
        ТабДок.Вывести (ОбластьШапка);
    КонецЦикла;
    ОбластьШапка = Макет.ПолучитьОбласть ("ШапкаГрафик");
    ТабДок.Вывести (ОбластьШапка);
    ТабДок.ВывестиГоризонтальныйРазделительСтраниц ();
    ОбластьОбязЧасть = Макет.ПолучитьОбласть ("ОбязЧасть");
    ТабДок.Вывести (ОбластьОбязЧасть);
    ОбластьДисцОбщейПодгОбязЧасти =
Макет.ПолучитьОбласть ("ДисцОбщейПодгОбязЧасти");
    ТабДок.Вывести (ОбластьДисцОбщейПодгОбязЧасти);
    КвоКредитов = 0; КвоОбъем = 0; КвоАудит = 0; КвоЛек = 0;
    КвоПракт = 0; КвоЛаб = 0; КвоСР = 0;
    Для Каждого ТекСтрока Из СсылкаНаДокумент.Обязательные Цикл
        Дисц = ТекСтрока.Дисциплина;

```

```

Если Дисц.Часть <> Справочники.ЧастиПлана.Обязательная
Тогда
    Продолжить;
КонецЕсли;
Если Дисц.Цикл <> Справочники.Циклы.ОбщаяПодготовка Тогда
    Продолжить;
КонецЕсли;
ОбластьВывода =
Макет.ПолучитьОбласть ("ОбластьДляЗаполнения");
ОбластьВывода.Параметры.Шифр = Дисц.Шифр;
ОбластьВывода.Параметры.НазвДисц = Дисц;
Если ТекСтрока.СеместровыйКонтроль =
Перечисления.ВидыСеместровогоКонтроля.Экзамен Тогда
    ОбластьВывода.Параметры.КолЭкз =
ТекСтрока.НомерСеместра.Номер;
    ОбластьВывода.Параметры.КолЗач = "";
ИначеЕсли ТекСтрока.СеместровыйКонтроль =
Перечисления.ВидыСеместровогоКонтроля.Зачет Тогда
    ОбластьВывода.Параметры.КолЭкз = "";
    ОбластьВывода.Параметры.КолЗач =
ТекСтрока.НомерСеместра.Номер;
КонецЕсли;
Если ТекСтрока.СеместровыеРаботы =
Перечисления.ВидыСеместровыхРабот.КР Тогда
    ОбластьВывода.Параметры.КолКР =
ТекСтрока.НомерСеместра.Номер;
ИначеЕсли ТекСтрока.СеместровыеРаботы =
Перечисления.ВидыСеместровыхРабот.РГР Тогда
    ОбластьВывода.Параметры.РГР =
ТекСтрока.НомерСеместра.Номер;
КонецЕсли;
ОбластьВывода.Параметры.КредЕКТС = ТекСтрока.Кредиты;
ОбластьВывода.Параметры.ОбщОбъем = ТекСтрока.ОбщийОбъем;
ОбластьВывода.Параметры.ВсегоАудит =
ТекСтрока.ВсегоАудиторные;
ОбластьВывода.Параметры.Лекции = ТекСтрока.Лекций;
ОбластьВывода.Параметры.Практич = ТекСтрока.Практик;
ОбластьВывода.Параметры.Лаб = ТекСтрока.Лабораторных;
ОбластьВывода.Параметры.СР = ТекСтрока.СРС;
ТабДок.Вывести (ОбластьВывода);
КвоКредитов = КвоКредитов + ТекСтрока.Кредиты;
КвоОбъем = КвоОбъем + ТекСтрока.ОбщийОбъем;
КвоАудит = КвоАудит + ТекСтрока.ВсегоАудиторные;
КвоЛек = КвоЛек + ТекСтрока.Лекций;
КвоПракт = КвоПракт + ТекСтрока.Практик;
КвоЛаб = КвоЛаб + ТекСтрока.Лабораторных;
КвоСР = КвоСР + ТекСтрока.СРС;
КонецЦикла;

ОбластьВывода = Макет.ПолучитьОбласть ("ВсегоЦикл_11");
ОбластьВывода.Параметры.СуммаКредЕКТС = КвоКредитов;
ОбластьВывода.Параметры.СуммаОбщОбъемЧас = КвоОбъем;
ОбластьВывода.Параметры.СуммаВсегоАудитЧас = КвоАудит;

```

```

ОбластьВывода.Параметры.СуммаЛекАудитЧас = КвоЛек;
ОбластьВывода.Параметры.СуммаПрактичАудитЧас = КвоПракт;
ОбластьВывода.Параметры.СуммаЛабаАудитЧас = КвоЛаб;
ОбластьВывода.Параметры.СуммаСамостЧас = КвоСР;
ТабДок.Вывести(ОбластьВывода);
ОбластьДисцПрофПодгОбязЧасть =
Макет.ПолучитьОбласть("ДисцПрофПодгОбязЧасть");
ТабДок.Вывести(ОбластьДисцПрофПодгОбязЧасть);
КвоКредитов = 0; КвоОбъем = 0; КвоАудит = 0; КвоЛек = 0;
КвоПракт = 0; КвоЛаб = 0; КвоСР = 0;
Для Каждого ТекСтрока Из СсылкаНаДокумент.Обязательные Цикл
    Дисц = ТекСтрока.Дисциплина;
    Если Дисц.Часть <> Справочники.ЧастиПлана.Обязательная
Тогда
        Продолжить;
    КонецЕсли;
    Если Дисц.Специальность = Справочники.Специальности.С125
Тогда
        Если Дисц.Цикл <> Справочники.Циклы.ПрофПодготовка
Тогда
            Продолжить;
        КонецЕсли;
        ИначеЕсли Дисц.Специальность =
Справочники.Специальности.С122 Тогда
            Если Дисц.Цикл <> Справочники.Циклы.СпецПодготовка
Тогда
                Продолжить;
            КонецЕсли;
        КонецЕсли;
        ОбластьВывода =
Макет.ПолучитьОбласть("ОбластьДляЗаполнения");
        ОбластьВывода.Параметры.Шифр = Дисц.Шифр;
        ОбластьВывода.Параметры.НазвДисц = Дисц;
        Если ТекСтрока.СеместровыйКонтроль =
Перечисления.ВидыСеместровогоКонтроля.Экзамен Тогда
            ОбластьВывода.Параметры.КолЭкз =
ТекСтрока.НомерСеместра.Номер;
            ОбластьВывода.Параметры.КолЗач = "";
            ИначеЕсли ТекСтрока.СеместровыйКонтроль =
Перечисления.ВидыСеместровогоКонтроля.Зачет Тогда
                ОбластьВывода.Параметры.КолЭкз = "";
                ОбластьВывода.Параметры.КолЗач =
ТекСтрока.НомерСеместра.Номер;
            КонецЕсли;
            Если ТекСтрока.СеместровыеРаботы =
Перечисления.ВидыСеместровыхРабот.КР Тогда
                ОбластьВывода.Параметры.КолКР =
ТекСтрока.НомерСеместра.Номер;
            ИначеЕсли ТекСтрока.СеместровыеРаботы =
Перечисления.ВидыСеместровыхРабот.РГР Тогда
                ОбластьВывода.Параметры.РГР =
ТекСтрока.НомерСеместра.Номер;
            КонецЕсли;

```

```

        ОбластьВывода.Параметры.КредЕКТС = ТекСтрока.Кредиты;
        ОбластьВывода.Параметры.ОбщОбъем = ТекСтрока.ОбщийОбъем;
        ОбластьВывода.Параметры.ВсегоАудит =
ТекСтрока.ВсегоАудиторные;
        ОбластьВывода.Параметры.Лекции = ТекСтрока.Лекций;
        ОбластьВывода.Параметры.Практич = ТекСтрока.Практик;
        ОбластьВывода.Параметры.Лаб = ТекСтрока.Лабораторных;
        ОбластьВывода.Параметры.СР = ТекСтрока.СРС;
        ТабДок.Вывести(ОбластьВывода);
        КвоКредитов = КвоКредитов + ТекСтрока.Кредиты;
        КвоОбъем = КвоОбъем + ТекСтрока.ОбщийОбъем;
        КвоАудит = КвоАудит + ТекСтрока.ВсегоАудиторные;
        КвоЛек = КвоЛек + ТекСтрока.Лекций;
        КвоПракт = КвоПракт + ТекСтрока.Практик;
        КвоЛаб = КвоЛаб + ТекСтрока.Лабораторных;
        КвоСР = КвоСР + ТекСтрока.СРС;
    КонечЦикла;
    ОбластьВывода =Макет.ПолучитьОбласть("ВсегоЦикл_12");
    ОбластьВывода.Параметры.СуммаКредЕКТС = КвоКредитов;
    ОбластьВывода.Параметры.СуммаОбщОбъемЧас = КвоОбъем;
    ОбластьВывода.Параметры.СуммаВсегоАудитЧас = КвоАудит;
    ОбластьВывода.Параметры.СуммаЛекАудитЧас = КвоЛек;
    ОбластьВывода.Параметры.СуммаПрактичАудитЧас = КвоПракт;
    ОбластьВывода.Параметры.СуммаЛабАудитЧас = КвоЛаб;
    ОбластьВывода.Параметры.СуммаСамостЧас = КвоСР;
    ТабДок.Вывести(ОбластьВывода);
КонечПроцедуры

//код програми для документу робочого навчального плану
&НаКлиенте
Процедура ПрограмаПриИзменении(Элемент)
    Объект.Специальность = Специальность(Объект.Программа);
КонечПроцедуры
&НаСервере
Функция Специальность(ВыбПрограмма)
    Возврат ОбщегоНазначения.ПолучитьСпециальность(ВыбПрограмма);
КонечФункции
&НаКлиенте
Процедура УчебныеДисциплиныДисциплинаПриИзменении(Элемент)
    ТекущиеДанные = Элементы.УчебныеДисциплины.ТекущиеДанные;
    ТекущиеДанные.Шифр = Шифр(ТекущиеДанные.Дисциплина);
КонечПроцедуры
&НаСервере
Функция Шифр(ВыбДисциплина)
    Возврат ОбщегоНазначения.ПолучитьШифрДисциплины(ВыбДисциплина);
КонечФункции

&НаКлиенте
Процедура ЗаполнитьДисциплинами(Команда)
    спис = Новый СписокЗначений(); //спис.Добавить(
    спис.Добавить("ЗаполнитьВсе", "Все дисциплины",);
    спис.Добавить("ЗаполнитьОбязательные", "Обязательные
дисциплины",);

```

```

    спис.Добавить ("ЗаполнитьВыборочные", "Выборочные
дисциплины", , );
    ВыбЭлемент = спис.ВыбратьЭлемент();
    Если ВыбЭлемент = Неопределено Тогда
        Возврат;
    Иначе
        Если ВыбЭлемент.Значение = "ЗаполнитьВсе" Тогда
            ЗаполнитьВсеДисциплины();
        ИначеЕсли ВыбЭлемент.Значение = "ЗаполнитьОбязательные"
Тогда
            ЗаполнитьОбязательные();
        ИначеЕсли ВыбЭлемент.Значение = "ЗаполнитьВыборочные" Тогда
            ЗаполнитьВыборочные();
        КонецЕсли;
    КонецЕсли;
КонецПроцедуры
&НаКлиенте
Процедура ЗаполнитьВсеДисциплины()
    спис = Новый СписокЗначений();
    СформироватьЗапросПоДисциплинам(спис, "Все");
    Объект.Обязательные.Очистить();
    Объект.Выборочные.Очистить();
    ЗаполнитьТабличнуюЧастьОбязательные(спис);
    ЗаполнитьТабличнуюЧастьВыборочные(спис);
КонецПроцедуры
&НаКлиенте
Процедура ЗаполнитьОбязательные()
    спис = Новый СписокЗначений();
    СформироватьЗапросПоДисциплинам(спис, "Обязательные");
КонецПроцедуры
&НаКлиенте
Процедура ЗаполнитьВыборочные()
    спис = Новый СписокЗначений();
    СформироватьЗапросПоДисциплинам(спис, "Выборочные");
КонецПроцедуры
&НаСервере
Процедура СформироватьЗапросПоДисциплинам(спис, видДисциплины)
    Запрос = Новый Запрос;
    Запрос.Текст =
        "ВЫБРАТЬ
        | СоставДисциплиныСрезПоследних.Специальность КАК
Специальность,
        | СоставДисциплиныСрезПоследних.Программа КАК Программа,
        | СоставДисциплиныСрезПоследних.Уровень КАК Уровень,
        | СоставДисциплиныСрезПоследних.Дисциплина КАК
Дисциплина,
        | СоставДисциплиныСрезПоследних.УчебныйГод КАК
УчебныйГод,
        | СоставДисциплиныСрезПоследних.НомерСеместра КАК
НомерСеместра,
        | СоставДисциплиныСрезПоследних.СеместровыйКонтроль КАК
СеместровыйКонтроль,

```



```

        | СоставДисциплиныСрезПоследних.СеместровыеРаботы КАК
СеместровыеРаботы,
        | СоставДисциплиныСрезПоследних.Кредиты КАК Кредиты,
        | СоставДисциплиныСрезПоследних.ОбщийОбъем КАК
ОбщийОбъем,
        | СоставДисциплиныСрезПоследних.Лекций КАК Лекций,
        | СоставДисциплиныСрезПоследних.Практик КАК Практик,
        | СоставДисциплиныСрезПоследних.Лабораторных КАК
Лабораторных,
        | СоставДисциплиныСрезПоследних.СРС КАК СРС
        | ИЗ
        | РегистрСведений.СоставДисциплины.СрезПоследних КАК
СоставДисциплиныСрезПоследних
        | ГДЕ
        | СоставДисциплиныСрезПоследних.УчебныйГод = &УчГод
        | И СоставДисциплиныСрезПоследних.Программа = &Программа
        | И СоставДисциплиныСрезПоследних.Специальность =
&Специальность
        | И СоставДисциплиныСрезПоследних.ФормаОбучения =
&ФормаОбуч
        | И СоставДисциплиныСрезПоследних.Уровень = &Уровень
        | И СоставДисциплиныСрезПоследних.НомерСеместра =
&НомерСеместра
        |
        | СГРУППИРОВАТЬ ПО
        | СоставДисциплиныСрезПоследних.Специальность,
        | СоставДисциплиныСрезПоследних.Программа,
        | СоставДисциплиныСрезПоследних.Уровень,
        | СоставДисциплиныСрезПоследних.УчебныйГод,
        | СоставДисциплиныСрезПоследних.НомерСеместра,
        | СоставДисциплиныСрезПоследних.Дисциплина,
        | СоставДисциплиныСрезПоследних.СеместровыйКонтроль,
        | СоставДисциплиныСрезПоследних.СеместровыеРаботы,
        | СоставДисциплиныСрезПоследних.Кредиты,
        | СоставДисциплиныСрезПоследних.ОбщийОбъем,
        | СоставДисциплиныСрезПоследних.Лекций,
        | СоставДисциплиныСрезПоследних.Практик,
        | СоставДисциплиныСрезПоследних.Лабораторных,
        | СоставДисциплиныСрезПоследних.СРС";

Запрос.УстановитьПараметр ("Программа", Объект.Программа);
Запрос.УстановитьПараметр ("Специальность",
Объект.Специальность);
Запрос.УстановитьПараметр ("Уровень", Объект.Уровень);
Запрос.УстановитьПараметр ("УчГод", Объект.УчебныйГод);
Запрос.УстановитьПараметр ("ФормаОбуч", Объект.ФормаОбучения);
Запрос.УстановитьПараметр ("Уровень", Объект.Уровень);
Запрос.УстановитьПараметр ("НомерСеместра", Объект.Семестр);
РезультатЗапроса = Запрос.Выполнить ();
ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать ();
Если видДисциплины = "Обязательные" Тогда
    Объект.Обязательные.Очистить ();
ИначеЕсли видДисциплины = "Выборочные" Тогда

```

```

    Объект.Выборочные.Очистить ();
    КонецЕсли;
    Пока ВыборкаДетальныеЗаписи.Следующий() Цикл
        Дисцип = ВыборкаДетальныеЗаписи.Дисциплина;
        Если видДисциплины = "Обязательные" Тогда
            Если Дисцип.Часть =
Справочники.ЧастиПлана.Обязательная Тогда
                НовСтрока = Объект.Обязательные.Добавить ();
                НовСтрока.Дисциплина = Дисцип;
                НовСтрока.Шифр = НовСтрока.Дисциплина.Шифр;
                НовСтрока.НомерСеместра =
ВыборкаДетальныеЗаписи.НомерСеместра;
                НовСтрока.СеместровыйКонтроль =
ВыборкаДетальныеЗаписи.СеместровыйКонтроль;
                НовСтрока.СеместровыеРаботы =
ВыборкаДетальныеЗаписи.СеместровыеРаботы;
                НовСтрока.Кредиты =
ВыборкаДетальныеЗаписи.Кредиты;
                НовСтрока.ОбщийОбъем =
ВыборкаДетальныеЗаписи.ОбщийОбъем;
                НовСтрока.Лекций = ВыборкаДетальныеЗаписи.Лекций;
                НовСтрока.Практик =
ВыборкаДетальныеЗаписи.Практик;
                НовСтрока.Лабораторных =
ВыборкаДетальныеЗаписи.Лабораторных;
                НовСтрока.СРС = ВыборкаДетальныеЗаписи.СРС;
                НовСтрока.ВсегоАудиторные = НовСтрока.Лекций +
НовСтрока.Практик + НовСтрока.Лабораторных;
            КонецЕсли;
            ИначеЕсли видДисциплины = "Выборочные" Тогда
                Если Дисцип.Часть = Справочники.ЧастиПлана.Выборочная
Тогда
                    НовСтрока = Объект.Выборочные.Добавить ();
                    НовСтрока.Дисциплина = Дисцип;
                    НовСтрока.Шифр = НовСтрока.Дисциплина.Шифр;
                    НовСтрока.НомерСеместра =
ВыборкаДетальныеЗаписи.НомерСеместра;
                    НовСтрока.СеместровыйКонтроль =
ВыборкаДетальныеЗаписи.СеместровыйКонтроль;
                    НовСтрока.СеместровыеРаботы =
ВыборкаДетальныеЗаписи.СеместровыеРаботы;
                    НовСтрока.Кредиты =
ВыборкаДетальныеЗаписи.Кредиты;
                    НовСтрока.ОбщийОбъем =
ВыборкаДетальныеЗаписи.ОбщийОбъем;
                    НовСтрока.Лекций = ВыборкаДетальныеЗаписи.Лекций;
                    НовСтрока.Практик =
ВыборкаДетальныеЗаписи.Практик;
                    НовСтрока.Лабораторных =
ВыборкаДетальныеЗаписи.Лабораторных;
                    НовСтрока.СРС = ВыборкаДетальныеЗаписи.СРС;
                    НовСтрока.ВсегоАудиторные = НовСтрока.Лекций +
НовСтрока.Практик + НовСтрока.Лабораторных;

```

```

        КонечЕсли;
    КонечЕсли;
    КонечЦикла;
КонечПроцедуры
&НаСервере
Процедура СоздатьТаблицуЗначенийСоставДисциплины (тз)
    тз.Колонки.Добавить ("НомСем");
    тз.Колонки.Добавить ("Контроль");
    тз.Колонки.Добавить ("Работы");
    тз.Колонки.Добавить ("Кредиты");
    тз.Колонки.Добавить ("ОбщийОбъем");
    тз.Колонки.Добавить ("Лекций");
    тз.Колонки.Добавить ("Практик");
    тз.Колонки.Добавить ("Лаб");
    тз.Колонки.Добавить ("СРС");
КонечПроцедуры
&НаСервере
Процедура ЗаполнитьТабличнуюЧастьОбязательные (спис)
    тз = Новый ТаблицаЗначений;
    СоздатьТаблицуЗначенийСоставДисциплины (тз);
    Для к = 0 По спис.Количество() - 1 Цикл
        Дисцип = спис.Получить (к).Значение;
        Если Дисцип.Часть = Справочники.ЧастиПлана.Обязательная
Тогда
            тз.Очистить ();
            ЗаполнитьИзСоставаДисциплины (Дисцип, тз);
            Если тз.Количество() = 0 Тогда
                НовСтрока = Объект.Обязательные.Добавить ();
                НовСтрока.Дисциплина = Дисцип;
                НовСтрока.Шифр = НовСтрока.Дисциплина.Шифр;
            Иначе
                Для Каждого ТекСтрока Из тз Цикл
                    Если НЕ ЗначениеЗаполнено (Объект.Курс) Тогда
                        НовСтрока =
Объект.Обязательные.Добавить ();
                        НовСтрока.Дисциплина = Дисцип;
                        НовСтрока.Шифр =
НовСтрока.Дисциплина.Шифр;
                        НовСтрока.НомерСеместра =
ТекСтрока.НомСем;
                        НовСтрока.СеместровыйКонтроль =
ТекСтрока.Контроль;
                        НовСтрока.СеместровыеРаботы =
ТекСтрока.Работы;
                        НовСтрока.Кредиты = ТекСтрока.Кредиты;
                        НовСтрока.ОбщийОбъем =
ТекСтрока.ОбщийОбъем;
                        НовСтрока.Лекций = ТекСтрока.Лекций;
                        НовСтрока.Практик = ТекСтрока.Практик;
                        НовСтрока.Лабораторных = ТекСтрока.Лаб;
                        НовСтрока.СРС = ТекСтрока.СРС;
                        НовСтрока.ВсегоАудиторные =
НовСтрока.Лекций + НовСтрока.Практик + НовСтрока.Лабораторных;

```

```

//ИначеЕсли
    КонечЕсли;
    КонечЦикла;
    КонечЕсли;
    КонечЕсли;
    КонечЦикла;
    Объект.Обязательные.Сортировать ("Шифр");
КонечПроцедуры
&НаСервере
Процедура ЗаполнитьТабличнуюЧастьВыборочные (спис)
    тз = Новый ТаблицаЗначений;
    СоздатьТаблицуЗначенийСоставДисциплины(тз);
    Для к = 0 По спис.Количество() - 1 Цикл
        Дисцип = спис.Получить(к).Значение;
        тз.Очистить();
        ЗаполнитьИзСоставаДисциплины(Дисцип, тз);
        Если Дисцип.Часть = Справочники.ЧастиПлана.Выборочная Тогда
            тз.Очистить();
            ЗаполнитьИзСоставаДисциплины(Дисцип, тз);
            Если тз.Количество() = 0 Тогда
                НовСтрока = Объект.Выборочные.Добавить();
                НовСтрока.Дисциплина = Дисцип;
                НовСтрока.Шифр = НовСтрока.Дисциплина.Шифр;
            Иначе
                Для Каждого ТекСтрока Из тз Цикл
                    НовСтрока = Объект.Выборочные.Добавить();
                    НовСтрока.Дисциплина = Дисцип;
                    НовСтрока.Шифр = НовСтрока.Дисциплина.Шифр;
                    НовСтрока.НомерСеместра = ТекСтрока.НомСем;
                    НовСтрока.СеместровыйКонтроль =
ТекСтрока.Контроль;
                    НовСтрока.СеместровыеРаботы =
ТекСтрока.Работы;
                    НовСтрока.Кредиты = ТекСтрока.Кредиты;
                    НовСтрока.ОбщийОбъем = ТекСтрока.ОбщийОбъем;
                    НовСтрока.Лекций = ТекСтрока.Лекций;
                    НовСтрока.Практик = ТекСтрока.Практик;
                    НовСтрока.Лабораторных = ТекСтрока.Лаб;
                    НовСтрока.СРС = ТекСтрока.СРС;
                    НовСтрока.ВсегоАудиторные = НовСтрока.Лекций
+ НовСтрока.Практик + НовСтрока.Лабораторных;
                КонечЦикла;
            КонечЕсли;
        КонечЕсли;
    КонечЦикла;
    Объект.Выборочные.Сортировать ("Шифр");
КонечПроцедуры
&НаСервере
Процедура ЗаполнитьИзСоставаДисциплины(Дисцип, тз)
    Запрос = Новый Запрос;
    Запрос.Текст =

```

```

        "ВЫБРАТЬ
        |      СоставДисциплиныСрезПоследних.Дисциплина КАК
Дисциплина,
        |      СоставДисциплиныСрезПоследних.НомерСеместра КАК
НомерСеместра,
        |      СоставДисциплиныСрезПоследних.СеместровыйКонтроль КАК
СеместровыйКонтроль,
        |      СоставДисциплиныСрезПоследних.СеместровыеРаботы КАК
СеместровыеРаботы,
        |      СоставДисциплиныСрезПоследних.Кредиты КАК Кредиты,
        |      СоставДисциплиныСрезПоследних.ОбщийОбъем КАК
ОбщийОбъем,
        |      СоставДисциплиныСрезПоследних.Лекций КАК Лекций,
        |      СоставДисциплиныСрезПоследних.Практик КАК Практик,
        |      СоставДисциплиныСрезПоследних.Лабораторных КАК
Лабораторных,
        |      СоставДисциплиныСрезПоследних.СРС КАК СРС
        |ИЗ
        |
        РегистрСведений.СоставДисциплины.СрезПоследних (&ВыбДата,
Дисциплина = &ВыбДисц) КАК СоставДисциплиныСрезПоследних";
        Запрос.УстановитьПараметр ("ВыбДата", Объект.Дата);
        Запрос.УстановитьПараметр ("ВыбДисц", Дисцип);
        РезультатЗапроса = Запрос.Выполнить ();
        ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать ();
        Пока ВыборкаДетальныеЗаписи.Следующий () Цикл
            НовСтрока = тз.Добавить ();
            НовСтрока.НомСем = ВыборкаДетальныеЗаписи.НомерСеместра;
            НовСтрока.Контроль =
ВыборкаДетальныеЗаписи.СеместровыйКонтроль;
            НовСтрока.Работы =
ВыборкаДетальныеЗаписи.СеместровыеРаботы;
            НовСтрока.Кредиты = ВыборкаДетальныеЗаписи.Кредиты;
            НовСтрока.ОбщийОбъем = ВыборкаДетальныеЗаписи.ОбщийОбъем;
            НовСтрока.Лекций = ВыборкаДетальныеЗаписи.Лекций;
            НовСтрока.Практик = ВыборкаДетальныеЗаписи.Практик;
            НовСтрока.Лаб = ВыборкаДетальныеЗаписи.Лабораторных;
            НовСтрока.СРС = ВыборкаДетальныеЗаписи.СРС;
        КонецЦикла;
    КонецПроцедуры
&НаКлиенте
Процедура УчебныеДисциплиныПриОкончанииРедактирования (Элемент,
НоваяСтрока, ОтменаРедактирования)
    ТекущиеДанные = Элементы.УчебныеДисциплины.ТекущиеДанные;
    ТекущиеДанные.ВсегоАудиторные = ТекущиеДанные.Лекций +
ТекущиеДанные.Практик + ТекущиеДанные.Лабораторных;
    ТекущиеДанные.ОбщийОбъем = ТекущиеДанные.Лекций +
ТекущиеДанные.Практик + ТекущиеДанные.Лабораторных +
ТекущиеДанные.СРС;
    КонецПроцедуры
&НаКлиенте
Процедура ОбязательныеПриОкончанииРедактирования (Элемент,
НоваяСтрока, ОтменаРедактирования)

```

```

    ТекущиеДанные = Элементы.Обязательные.ТекущиеДанные;
    ТекущиеДанные.ВсегоАудиторные = ТекущиеДанные.Лекций +
ТекущиеДанные.Практик + ТекущиеДанные.Лабораторных;
    ТекущиеДанные.ОбщийОбъем = ТекущиеДанные.Лекций +
ТекущиеДанные.Практик + ТекущиеДанные.Лабораторных +
ТекущиеДанные.СРС;
КонецПроцедуры
&НаКлиенте
Процедура ВыборочныеПриОкончанииРедактирования (Элемент, НоваяСтрока,
ОтменаРедактирования)
    ТекущиеДанные = Элементы.Выборочные.ТекущиеДанные;
    ТекущиеДанные.ВсегоАудиторные = ТекущиеДанные.Лекций +
ТекущиеДанные.Практик + ТекущиеДанные.Лабораторных;
    ТекущиеДанные.ОбщийОбъем = ТекущиеДанные.Лекций +
ТекущиеДанные.Практик + ТекущиеДанные.Лабораторных +
ТекущиеДанные.СРС;
КонецПроцедуры
&НаКлиенте
Процедура КурсПриИзменении (Элемент)
    ЗаполнитьСписокСеместрамиВыбранногоКурса (Объект.Курс);
КонецПроцедуры
&НаСервере
Процедура ЗаполнитьСписокСеместрамиВыбранногоКурса (ВыбКурс)
    ТекСеместр.Вставить (0, ВыбКурс.СеместрОсенний);
    ТекСеместр.Вставить (1, ВыбКурс.СеместрВесенний);
КонецПроцедуры
&НаКлиенте
Процедура ВыборСеместраИзСписка (Команда)
    ВыбЭлемент = "";
    ВыбЭлемент = ТекСеместр.ВыбратьЭлемент ("Оберіть семестр",
ВыбЭлемент);
    Если ВыбЭлемент <> Неопределено Тогда
        Объект.Семестр = ВыбЭлемент.Значение;
    КонецЕсли;
КонецПроцедуры

```