**Добрый день, 12 группа!**

Продолжаем общаться дистанционно.

Сегодня мы разберем компьютерные модели различных процессов

Задать вопросы, а также прислать ответы вы можете

1. на адрес электронной почты: ddrmx@ya.ru
2. через соцсеть <https://vk.com/ddrmx>

С уважением, Максим Андреевич.

ЗАНЯТИЕ ПО ТЕМЕ:

Компьютерные модели различных процессов (MVC). (1 ЧАС)

MVC — это шаблон программирования, который позволяет разделить логику приложения на три части:

* **Model (модель)**. Получает данные от контроллера, выполняет необходимые операции и передаёт их в вид.
* **View (вид или представление)**. Получает данные от модели и выводит их для пользователя.
* **Controller (контроллер)**. Обрабатывает действия пользователя, проверяет полученные данные и передаёт их модели.

Может показаться, что это что-то запутанное, но на самом деле всё просто.

## Как работает MVC

Лучше всего понять концепцию MVC можно на реальном примере — ресторане с фастфудом. В нём посетители (пользователи) подходят к кассиру (одновременно вид и контроллер), видят меню и заказывают какое-нибудь блюдо.

Кассир проверяет, всё ли в порядке с заказом, и после оплаты передаёт нужные данные повару (модель). Повар готовит заказанное блюдо, хотя понятия не имеет о том, как выглядит посетитель, оплатил ли он заказ и так далее.

Когда модель закончит свою работу, она отправит результат в вид — обратно кассиру, который, в свою очередь, отдаст готовое блюдо посетителю.

Если же говорить о приложениях, то компоненты будут следующие:

* **Вид** — интерфейс.
* **Контроллер** — обработчик событий, инициируемых пользователем (нажатие на кнопку, переход по ссылке, отправка формы).
* **Модель** — метод, который запускается обработчиком и выполняет все основные операции (получение записей из базы данных, проведение вычислений).

Стоит также отметить, что реализация паттерна MVC может отличаться в зависимости от задачи. Например, в веб-разработке модель и вид взаимодействуют друг с другом через контроллер (как в примере с рестораном), а в приложениях модель может сама уведомлять вид, что нужно что-то изменить.







## Письменно ответьте на вопрос: Зачем программистам нужен MVC?

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ ПО ТЕМЕ:

Компьютерные модели. (1 ЧАС)

Код реализации паттерна MVC (Model View Controller):

Представление:

1. **namespace** MVC
2. {
3. **public** partial **class** View : Form
4. {
5. // создаем объект класса Controller
6. Controller controller = **new** Controller();
7. **public** View()
8. {
9. InitializeComponent();
10. }
11.
12. **private** **void** button1\_Click(**object** sender, EventArgs e)
13. {
14. **if** (textBox1.Text != "")
15. // выводим результат
16. MessageBox.Show(controller.Question(textBox1.Text));
17. }
18. }
19. }

Контроллер:

1. **namespace** MVC
2. {
3. **class** Controller
4. {
5. **public** **string** Question(**string** msg)//msg - то что ищем
6. {
7. Model model =  **new** Model();
8. **return** "На даный момент у нас товар "
9. + model.GetAnser(msg);
10. }
11. }
12. }

Модель:

1. **namespace** MVC
2. {
3. **class** Model
4. {
5. //question - то что ищем
6. **public** **string** GetAnser(**string** question)
7. {
8. **return** DataBase.GetAnser(question);
9. }
10. }
11. }

Класс помощник. Умеет общаться с базой данных:

1. **namespace** MVC
2. {
3. **static** **class** DataBase
4. {
5. //question - то что ищем
6. **public** **static** **string** GetAnser(**string** question)
7. {
8. **string** path = "D:\\shop.txt";// путь к базе данных ):
9.
10. // создаем объект (читатель)
11. **using** (StreamReader sr =
12. **new** StreamReader(path, Encoding.Default))
13. {
14. **string** str = **null**;
15. **while** (**true**)
16. {
17. //если пустая, то это конец
18. str = sr.ReadLine();
19. **if** (str != **null**)
20. {
21. //если считаная строчка
22. //равна строке для поиска
23. **if** (str == question)
24. **return** "есть в наличие.";
25. }
26. **else**
27. **return** "нет в наличие.";
28. }
29. }
30. }
31. }
32. }

Тестируем:



Видим, что в текстовом файле есть такая запись, по этому ответ следующий:



***Контрольные вопросы***

* 1. Что такое MVC?
	2. Опишите MVC в других языках программирования.