



Лист с заданиями к лекции №2

Теоретические задания

1. ООП
2. 4 принципа ООП
3. Класс; объект; составляющие объекта
4. Абстрактные классы и интерфейсы (изучить самостоятельно)
5. Диаграммы классов в UML (изучить самостоятельно)
6. Сильное сцепление

Источники:

<https://about.draw.io/uml-class-diagrams-in-draw-io/>

<https://book.uml3.ru/>

<https://www.intuit.ru/studies/courses/1007/229/lecture/5950>

Практические задания

Задания должны располагаться в отдельных папках, по типу task-[№ лекции]/[№ упражнения], например task-2/1, task-2/2. Диаграммы желательно экспортировать в картинки, например .jpeg.

В итоге для задания 3 файлы бы располагались по таким путям:

task-2/3/client.jpg

task-2/3/employee.jpg

Для каждой лекции отдельная ветка из master - Task-[№ лекции]. Для текущей будет Task-2. По завершению мерж-реквест в master на преподавателя.

Задание 1 (сложность 1)

Написать первую программу типа "Hello world" на Java. Программа должна состоять из одного класса с именем Greeting. Класс должен содержать один методом public static main(). Разработку класса выполнять в любом текстовом блокноте. Компиляцию проводить при помощи javac.

Требования к выполнению:

- 1) Исходный .java файл и откомпилированный .class файл должны быть «вкомитаны» в GIT
- 2) Использование для выполнения любых IDE не допускается
- 3) Для вывода сообщения использовать - System.out.println("Hello Java! ");

Задание 2 (сложность 1)

Найти и установить удобную для вас программу для рисования диаграмм классов (как вариант может быть использована бесплатная версия Visual Paradigm от Agilian -

<http://www.visual-paradigm.com/download/vpuml.jsp?edition=ce> или онлайн сервис

<https://www.draw.io/>).

Данный документ и любая информация, содержащаяся на нем, является интеллектуальной собственностью компании Senla. Распространение, копирование, цитирование, воспроизведение в любом виде допускается только с письменного разрешения Senla.

Задание 3 (сложность 2)

Составить детализированную диаграмму классов для сущности, которая будет использована в гипотетической программе (2 на выбор):

- 1) Сотрудник – сущность будет использована в программе для отдела кадров
- 2) Пациент — сущность будет использована в медицинской программе для поликлиник
- 3) Самолет — сущность будет использована в программе контроля полетов для диспетчерской аэропорта
- 4) Клиент – сущность будет использована в банковской системе работы с клиентами
- 5) Резюме – сущность будет использована в программе для онлайн поиска работы

Требования к выполнению:

- 1) Не менее 4 классов
- 2) Диаграмма содержит иерархии и связи
- 3) Все классы должны иметь свойства и методы
- 4) Исходный файл диаграмм (в формате выбранного вами редактора) должен быть «вкомитаны» в GIT
- 5) Диаграммы должны быть сохранены в графический файл и «вкомитаны» в GIT
- 6) Текст на диаграммах должен быть легко читабельным

Задание 4 (сложность 10) *

Составить детализированную диаграмму классов для простой программы-игры:

- 1) Lines
- 2) Тетрис
- 3) Сапер
- 4) Пасьянс
- 5) Арканойд *

Требования к выполнению:

- 1) Диаграмма должна покрывать максимальное количество аспектов реализации игры
- 2) Не менее 15 классов (должны встречаться интерфейсы и абстрактные классы)
- 3) Соблюдать принцип сильного сцепления
- 4) Диаграмма содержит иерархии
- 5) Все классы должны иметь свойства и методы
- 6) Диаграмма содержит агрегации и композиции
- 7) Исходный файл диаграмм (в формате выбранного вами редактора) должен быть «вкомитаны» в GIT
- 8) Диаграммы должны быть сохранены в графический файл и «вкомитаны» в GIT
- 9) Тексты на диаграммах должны быть легко читабельным

Данный документ и любая информация, содержащаяся на нем, является интеллектуальной собственностью компании Senla. Распространение, копирование, цитирование, воспроизведение в любом виде допускается только с письменного разрешения Senla.