Министерство образования и науки Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем

и программной инженерии

Лабораторная работа № 2

“ Начало работы с Java SE”

по дисциплине

«Платформонезависимое программирование»

Выполнил:

ст. гр. ПРИ-118

В. В. Баранов

Принял:

А. Т. Казаев

Владимир 2020

**Цель работы**

Освоить базовые принципы ООП. Получить практические навыки по работе с классами и объектами на платформе java. Научиться проводить декомпозицию приложений с использованием пакетов.

**Выполнение работы**

Независимо от варианта задания, необходимо выполнить следующее:

1. Cоздания приложения с пакетами

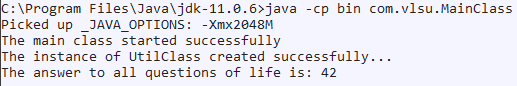
MainClass.java

1. **package** com**.**vlsu**;**
2. **import** com.vlsu.util.UtilClass**;**
3. **public** **class** MainClass
4. **{**
5. **public** **static** **void** main**(**String **[]** args**)**
6. **{**
7. System**.**out**.**println**(**"The main class started successfully"**);**
8. UtilClass uc **=** **new** UtilClass**();**
9. System**.**out**.**println**(**"The answer to all questions of life is: " **+** uc**.**getAnswer**());**
10. **}**
11. **}**

UtilClass.java

1. **package** com**.**vlsu**.**util**;**
2. **public** **class** UtilClass
3. **{**
4. **public** UtilClass**()**
5. **{**
6. System**.**out**.**println**(**"The instance of UtilClass created successfully..."**);**
7. **}**
8. **public** String getAnswer**()**
9. **{**
10. **return** "42"**;**
11. **}**
12. **}**

Тестирование и отладка:



*Рисунок 1. Результат работы программы*

2. Независимо от варианта, необходимо реализовать классы Rectangle и Point. Класс Point определяется двумя координатами точки, класс Rectangle определяется точками левого верхнего угла и правого нижнего.

Point.java

1. **package** figure**.**simple**;**
2. **public** **class** Point
3. **{**
4. **public** **float** x**;**
5. **public** **float** y**;**
6. **public** Point**(float** x**,** **float** y**)**
7. **{**
8. **this.**x **=** x**;**
9. **this.**y **=** y**;**
10. **}**
11. **}**

Rectangle.java

1. **package** figure**.**compound**.**square**;**
2. **import** figure.simple.Point**;**
3. **public** **class** Rectangle
4. **{**
5. **public** Point topLeftPoint**,** bottomRightPoint**;**
6. **public** Rectangle**(**Point topLeftPoint**,** Point bottomRightPoint**)**
7. **{**
8. **this.**topLeftPoint **=** topLeftPoint**;**
9. **this.**bottomRightPoint **=** bottomRightPoint**;**
10. **}**
11. **}**

3. Необходимо реализовать приложение, работающее с кругами. Первый вспомогательный класс - точка - определяется параметрами x и y - координатами на плоскости. Второй вспомогательный классс - круг - определяется параметрами p и R - точкой центра круга и радиусом. Помимо этого, у отрезка должен быть метод inside, который в качестве аргумента принимает точку. Он должен возвращать true, если точка находится внутри круга, и false в противном случае Основной класс должен запросить у пользователя координаты центра круга и его радиус, после чего создать объект круга. После этого он должен запросить у пользователя координаты точки и вывести ответ - находится ли данная точка внутри круга.

Point.java

1. **package** figure**.**simple**;**
2. **public** **class** Point
3. **{**
4. **public** **float** x**;**
5. **public** **float** y**;**
6. **public** Point**(float** x**,** **float** y**)**
7. **{**
8. **this.**x **=** x**;**
9. **this.**y **=** y**;**
10. **}**
11. **}**

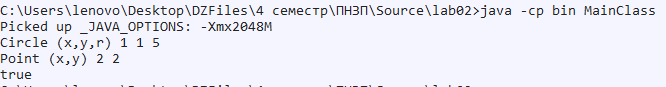
Circle.java

1. **package** figure**.**compound**;**
2. **import** figure.simple.Point**;**
3. **public** **class** Circle
4. **{**
5. **public** Point center = new Point(0, 0)**;**
6. **public** **float** radius**;**
7. **public** Circle**(float** x**,** **float** y**,** **float** radius**)**
8. **{**
9. center**.**x **=** x**;**
10. center**.**y **=** y**;**
11. **this.**radius **=** radius**;**
12. **}**
13. **public** **boolean** inside**(**Point point**)**
14. **{**
15. **return** **(**point**.**x **-** center**.**x**)** **\*** **(**point**.**x **-** center**.**x**)** **+** **(**point**.**y **-** center**.**y**)** **\*** **(**point**.**y **-** center**.**y**)** **<=** radius **\*** radius**;**
16. **}**
17. **}**

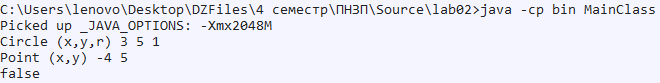
MainClass.java

1. **import** figure.compound.Circle**;**
2. **import** figure.simple.Point**;**
3. **import** java.util.Scanner**;**
4. **public** **class** MainClass
5. **{**
6. **public** **static** **void** main**(**String **[]** args**)**
7. **{**
8. Scanner console **=** **new** Scanner**(**System**.**in**);**
9. System**.**out**.**print**(**"Circle (x,y,r) "**);**
10. **float** x **=** console**.**nextFloat**();**
11. **float** y **=** console**.**nextFloat **();**
12. **float** r **=** console**.**nextFloat **();**
13. Circle circle **=** **new** Circle**(**x**,**y**,**r**);**
14. System**.**out**.**print**(**"Point (x,y) "**);**
15. x **=** console**.**nextFloat **();**
16. y **=** console**.**nextFloat **();**
17. Point point **=** **new** Point**(**x**,**y**);**
18. System**.**out**.**print**(**circle**.**inside**(**point**));**
19. **}**
20. **}**

Тестирование и отладка



*Рисунок 2. Рейзульта работы программы*



*Рисунок 3. Результа работы программы*

**Вывод**

В результате выполнения работы были освоены базовые принципы ООП. Получены практические навыки по работе с классами и объектами на платформе java. Получены навыки проведения декомпозиции приложений с использованием пакетов.