СПбНИУ ИТМО

Отчет

По лабораторной работе №1

«Программирование интернет приложений»

Выполнил: Ратников В.И

Группа 3105

Преподаватель: Гаврилов А.В.

Исходный код программы

**<HelloWorld.java>**

/\*\* Our basic class \*/

public class HelloWorld implements Iprintable

{

/\*\* Protected field with greeting \*/

protected String msg = "hello world";

/\*\* entry point to out main class \*/

public static void main(String[] args)

{

HelloWorld hw = new HelloWorld();

hw.flush();

hw.print();

Extender ext = new Extender("hello,eternal world");

ext.print();

String extracted\_field = ext.get\_field();

/\* we don't realy need that. but non-familiar classes coudn't get access to msg like on the next string \*/

System.out.println(ext.msg);

ext.flush();

ext.print();

}

/\*\* see Iprintable.print() \*/

@Override

public void print()

{

System.out.println( this.msg );

}

/\*\* see Iprintable.flush() \*/

@Override

public void flush()

{

this.msg = "";

}

/\*\* see Iprintable.get\_field() \*/

@Override

public String get\_field()

{

return this.msg;

}

/\*\* see Iprintable.set\_field(String) \*/

@Override

public void set\_field(String arg)

{

this.msg = arg;

}

}

**<Extender.java>**

/\*\* child class to HelloWorld \*/

class Extender extends HelloWorld implements Iprintable

{

/\*\* constructor to Extender class. Inherits msg field \*/

public Extender( String \_ms )

{

this.msg = \_ms;

}

/\*\* override of super.toString() \*/

@Override

public String toString()

{

return this.msg;

}

/\*\* see Iprintable.print() \*/

@Override

public void print()

{

System.out.println( this.msg );

}

/\*\* see Iprintable.flush() \*/

@Override

public void flush()

{

this.msg = "";

}

/\*\* see Iprintable.get\_field() \*/

@Override

public String get\_field()

{

return this.msg;

}

/\*\* see Iprintable.set\_field(String) \*/

@Override

public void set\_field(String arg)

{

this.msg = arg;

}

}

**<IPrintable.java>**

/\*\* contains abstract methods like print,flush,set\_field,get\_field \*/

public interface Iprintable

{

/\*\* print field \*/

public abstract void print();

/\*\* erases field value \*/

public abstract void flush();

/\*\* get field value \*/

public abstract String get\_field();

/\*\* set field value \*/

public abstract void set\_field(String arg);

}

Используемые утилиты

Java - интерпретатор java,запуск виртуальной машины jvm.

Пример : java –jar out.jar

Javac – компилятор. Преобразует исходный код в байт код.

Пример: javac \*.java

Javadoc – генератор документации на основе документирующих комментариев.

Пример javadod \*.java

Jar – архиватор

Пример : jar cfm out.jav MANIFEST.MF \*.class

Содержимое файла MANIFEST.MF  
Main-Class: HelloWorld

Выводы по работе:

Язык java сочетает в себе лучшие черты прочих языков программирования.

Переносимость, объектная ориентированности, отказоустойчивость.