# Overall Review and Exam Format

**Part 1 of 2:**

6:00 to 7:30

Open book

Do alone

Activities/Quizzes/ final Exam Part 1

Like a big quiz

**Part 2 of 2:**

7:30 – 9:30

Open book

IDE

Activities / Assignments / Final Exam Part 2

Like a small lab or assignment:

- pass some unit tests

- write some code

**Topics Review**

Class: A java file with data and behaviors that describe a general category

Object: One specific instance of a class

public class Student {} // This class can be used anywhere

class Student {} // this is not public – can be accessed from other classes in the same package only

private class Student {} // You can make a private constructor but not a private class

class badFirstNameException Exception {

BadFirstNameException(String m) {

super(m);

}

}

class Person {

public void hi() {

System.out.println(“Hi I am a person”);

}

}

class Student extends Person {

private final String firstName;

private double weightKg;

@override // this will check the signature it will check to see if you did it right

Public void hi() {

System.out.println(“Hi I am a student”);

}

public static final double LB\_PER\_KG = 2.2;

/\*\*

\* @param fName the first name

\* @throws BadFirstName Exception if fName is null or Blank

\*/

Student(final String fName) {

super();

If(fName == null || fName.isBlank()) {

throw new IllegalArgumentException(“bad first name”)

}

firstName = fName;

}

Public void setWeight(final int kg) {

this.weightKg = kg;

}

Public void setWeight(final double pounds) {

This.weightKg = pounds / LB\_PER\_KG;

}

}

class Main {

public static void main(final String[] args) {

List<Person> people;

// array of five strings

String[] strings;

strings = new String[5];

strings[0] = “hello”;

strings[3] = “goodbye”;

Strings[2] = “the end”;

// 1=>”one”, 2=>”two”

Map<Integer, String> numbers;

Numbers = new HashMap<>();

numbers.put(1, “one”);

numbers.put(2, “two”);

Set<Integer> keys;

keys = numbers.keySet();

for(Integer key:keys) {

System.out.println(numbers.get(key));

} // if you are dealing with collections you cant use Int

people = new ArrayList<>();

people.add(new Person());

// This one below will require a try/catch

try {

people.add(new Student(“Monika”));

System.out.println(“success”);

for(Person person: people) {

System.out.println(person);

person.hi(); // if there is a student hi it will print it if not then it will look in object and if not there it will not compile

}

Iterator<Person> it;

It = people.iterator();

While(it.hasNext()) {

Person p = it.next();

System.out.println(p);

}

} catch(final BadFirstNameException ex) {

System.out.println(ex.getMessage());

} finally {

// clean up actions – when you want things done regardless put it here

}

}

}

// read text from a file to print it

File file;

Scanner scanner;

String line;

File = new File(“whatever.txt”);

Scanner = new Scanner(file);

While(scanner.hasNextLine()) {

line = scanner.nextLine();

}