JAVAJAVA EXERCISE SHEET

TRY THE FOLLOWING EXERCISES TO GET BETTER AT JAVA!

Dr. Java and his accomplice both have a time travel machine that they use to travel to different eras in time in order to solve the problems of the people there using their programming skills. Through these exercises, Dr. Java will visit countless eras and using his different programming techniques will attempt to solve as many problems as he can.

1) In order to get the time travel machine to work we need to print the correct password. Try and print the following statement**: "Password: Dr. Java loves Dr. Who"**
Hint: use the println function.

   Now that we have access to the time machine, Dr. Java needs to set a destination to travel to first, out first stop is going to be the Egyptian period. Print the following statement: **"Set destination: Egypt"**. However for this exercise, you have to use a variable "destination" and set it to the place you want to travel to in order to print out the Egypt part.

   (This way you can easily change the destination just by changing the variable instead of having to change all the print statements in a large program).

Answers:

```
public class Exercise1 {
    public static void main (String args[]){

        System.out.println("Password: Dr. Java loves Dr. Who!");

        String destination = "Egypt";
        System.out.println("Set destination: " + destination);
    }
}
```

The time machine makes its way towards Cairo, the Egyptian capital, where Dr. Java stumbles across a group of people who are helping to build the pyramids. You notice that they are using pen and paper to note down the dimensions of the blocks they are using and have to note down the dimensions of each block by measuring the width, height and depth of each block and then they put down its volume. This takes a very long time so you decide that you will make the process quicker for them by writing a Java program.

2) First you need to write a class called BoxSpec that will include the attributes height, width and depth- think about what type they should be. The class will also need a method that will return the volume.

Next, you should make a constructor in this class that accepts the height, width and depth as parameters. This makes it easier to construct each box object since you don't have to assign the dimensions manually every time you produce a box- here you can just put it in as parameters.

Finally in the main() part of the program you should construct 3 box objects with the following dimensions:

Box 1: Height- 10, Width-5, Depth-3
Box 2: Height- 5, Width-15, Depth-2
Box 3: Height- 20, Width-5, Depth-5

Then you need to print out the volume using the volume method from the class BoxSpec.

Answers:

```
class BoxSpec {

    int height;
    int width;
    int depth;

    int volume(){
        return height*width*depth;
    }

    BoxSpec(int w, int h, int d){
        width = w;
        height = h;
        depth = d;
    }

}

public class Exercise2 {
    public static void main (String args[]){

        BoxSpec box1 = new BoxSpec(10,5,3);
        BoxSpec box2 = new BoxSpec(5,15,2);
        BoxSpec box3 = new BoxSpec(20,5,5);

        System.out.println("The volume of box1 is " + box1.volume());
        System.out.println("The volume of box2 is " + box2.volume());
        System.out.println("The volume of box3 is " + box3.volume());
    }
}
```

The Egyptians were surprised by how quickly they were able to get all the details down for all their blocks and as a result we are now ready to build the pyramids. However it is difficult to transport the blocks since they are so heavy and we forgot to take down the weight of each block! Luckily, the all of the blocks have the same density (3000 kg/m^3) so we can use the

equation "density = mass/ volume" in order to find the weight of each block. Write in another method into this class that will allow you to find the mass of the blocks and then print out the total mass of the 3 blocks individually and then combined in the main().

Answers:

```
class BoxSpec {

    int height;
    int width;
    int depth;

    int volume(){
        return height*width*depth;
    }

    BoxSpec(int w, int h, int d){
        width = w;
        height = h;
        depth = d;
    }

    int mass(){
       return volume()*3000;
    }

}

public class Exercise2 {
    public static void main (String args[]){

        BoxSpec box1 = new BoxSpec(2,3,5);
        BoxSpec box2 = new BoxSpec(5,15,2);
        BoxSpec box3 = new BoxSpec(20,5,5);


        System.out.println("The volume of box1 is " + box1.volume() + " metres
cubed");
        System.out.println("The mass of box1 is " + box1.mass() + " kg");

        System.out.println("The volume of box2 is " + box2.volume());
        System.out.println("The mass of box2 is " + box2.mass() + " kg");

        System.out.println("The volume of box3 is " + box3.volume());
        System.out.println("The mass of box3 is " + box2.mass() + " kg");

        System.out.println("The total mass combined is " + (box1.mass() +
box2.mass() + box3.mass()));
    }
}
```

Here you can see why it was useful to have a constructor and BoxSpec class because then with all the details about the box, even after we forgot to find the weight, we could use the stored data to find out the weights of the boxes using another method.

Now that Dr. Java was able to help out the Egyptians, he has decided to travel forwards in time to the Victorian Era. When we arrive in modern day England, we come across an odd scruffy looking man, upon speaking to him a little more, we find out he is actually a mathematician. He is has lost all of his working out and Maths which he was going to present to the Queen! There is not much time for him to do all his working out again, so Dr. Java decides to help him out using a little Java!

3) We need to write a program which has a Maths class, which contains methods that lets you square and cube a number.

   The next method is going to find the hypotenuse of a triangle and will take two parameters, the two shortest sides of a right-angled triangle and then it should return the hypotenuse of the triangle. Try use your square function that you used earlier to make this part shorter. In Java to get the square root, you use Math.sqrt().

   Now in the main, find the square and cube of the following numbers: 5, 10, 20,
   Next in the main, find the hypotenuse of the following triangles with side lengths: (3,4), (12,13) and (200,500).

```
class Maths{

    int square(int i){
        return i*i;
    }

    int cube(int i){
        return i*i*i;
    }

    double pythag(int a, int b){
        return Math.sqrt(square(a) + square(b));
    }

}

public class Exercise3 {
    public static void main (String args[]){

       Maths number = new Maths();

        System.out.println(number.square(3));
        System.out.println(number.square(5));
        System.out.println(number.square(20));

        System.out.println(number.cube(3));
        System.out.println(number.cube(5));
        System.out.println(number.cube(20));

        System.out.println(number.pythag(3,4));
        System.out.println(number.pythag(12,13));
        System.out.println(number.pythag(200,500));
    }
}
```

This way, the mathematician was able to put in any number he wanted and could instantly find the answers to any of his problems. If he wanted, he could write the code for more complicated equations and all he would have to do was put in the correct numbers and Java would do all the calculations for him. The man was very happy and thanked you as he made his way to meet the Queen.

Dr. Java is really tired having helped people in two different eras all in the space of one day. He decides he wants to have something to eat but prefers to have authentic Greek food, so he travels back in time to Ancient Greece. When he finds somewhere that serves food, he finds there are loads of people who need to have their orders taken and there is only one waiter to tend to everyone! Dr. Java is way too hungry to wait for the waiter to come around to him, so he writes a program that will take orders!

4) You need to write a program that will take user input and ask them for their name and their table number and will take their order. Each order will consist of a starter, main and dessert. The end of the program needs to print out their name and table number on one line and then on the line below it must print back their order.

   You will need to import java.util.Scanner at the top of the program to make use of scanner for user IO.

Answers:

```
import java.util.Scanner;

public class UserInput {
    public static void main (String args[]){

        Scanner user_input = new Scanner(System.in);

        System.out.print("Enter your first name: ");
        String first_name = user_input.next();

        System.out.print("Enter your table number: ");
        int table_number = user_input.nextInt();

        System.out.print("What do you want for starters: ");
        String starter = user_input.next();

        System.out.print("What do you want for the main course: ");
        String main = user_input.next();

        System.out.print("What do you want for dessert: ");
        String dessert = user_input.next();

        System.out.println("Order summary for " + first_name + " at table number "
+ table_number + " is: ");
        System.out.println(starter + ", " + main + ", " + dessert );
    }
}
```