1. Create a class Box that uses a parameterized method to initialize the dimensions of a box.(dimensions are width, height, depth of double type). The class should have a method that can return volume. Obtain an object and print the corresponding volume in main() function.

2. Create a new class called “Calculator” which contains the following:

1. A static method called powerInt(int num1,int num2) that accepts two integers and returns num1 to the power of num2 (num1 power num2).

2. A static method called powerDouble(double num1,int num2) that accepts one double and one integer and returns num1 to the power of num2 (num1 power num2).

3. Call your method from another class without instantiating the class (i.e. call it like Calculator.powerInt(12,10) since your methods are defined to be static)

Hint: Use Math.pow(double,double) to calculate the power.

3 Design a class that can be used by a health care professional to keep track of a patient’s vital statistics. Here’s what the class should do:

1. Construct a class called Patient

2. Store a String name for the patient

3. Store weight and height for patient as doubles

4. Construct a new patient using these values

5. Write a method called BMI which returns the patient’s BMI as a double. BMI can be calculated as BMI = ( Weight in Pounds / ( Height in inches x Height in inches ) ) x 703

6. Next, construct a class called “Patients” and create a main method. Create a Patient object and assign some height and weight to that object. Display the BMI of that patient.

Create a class called Author is designed as follows:

It contains:

• Three private instance variables: name (String), email (String), and gender (char of either ‘m’ or ‘f’).

• One constructor to initialize the name, email and gender with the given values.

And, a class called Book is designed as follows:

It contains:

• Four private instance variables: name (String), author (of the class Author you have just created), price (double), and qtyInStock (int). Assuming that each book is written by one author.

• One constructor which constructs an instance with the values given.

• Getters and setters: getName(), getAuthor(), getPrice(), setPrice(), getQtyInStock(), setQtyInStock(). Again there is no setter for name and author.

Write the class Book (which uses the Author class written earlier).

Try:

1. Printing the book name, price and qtyInStock from a Book instance. (Hint: aBook.getName())

2. After obtaining the “Author” object, print the Author (name, email & gender) of the book.

1. reate a class named ‘Animal’ which includes methods like eat() and sleep().

Create a child class of Animal named ‘Bird’ and override the parent class methods. Add a new method named fly().

Create an instance of Animal class and invoke the eat and sleep methods using this object.Create an instance of Bird class and invoke the eat, sleep and fly methods using this object.

Create a class called Person with a member variable name. Save it in a file called Person.java

2.c Create a class called Employee who will inherit the Person class.The other data members of the employee class are annual salary (double), the year the employee started to work, and the national insurance number which is a String.Save this in a file called Employee.java

Your class should have a reasonable number of constructors and accessor methods.

Write another class called TestEmployee, containing a main method to fully test your class definition.

3.A HighSchool application has two classes: the Person superclass and the Student subclass. Using inheritance, in this lab you will create two new classes, Teacher and CollegeStudent. A Teacher will be like Person but will have additional properties such as salary (the amount the teacher earns) and subject (e.g. “Computer Science”, “Chemistry”, “English”, “Other”). The CollegeStudent class will extend the Student class by adding a year (current level in college) and major (e.g. “Electrical Engineering”, “Communications”, “Undeclared”).

1. Create a base class Fruit which has name ,taste and size as its attributes. A method called eat() is created which describes the name of the fruit and its taste. Inherit the same in 2 other class Apple and Orange and override the eat() method to represent each fruit taste.

2. Write a program to create a class named shape. It should contain 2 methods- draw() and erase() which should print “Drawing Shape” and “Erasing Shape” respectively.

For this class we have three sub classes- Circle, Triangle and Square and each class override the parent class functions- draw () and erase ().

The draw() method should print “Drawing Circle”, “Drawing Triangle”, “Drawing Square” respectively.

The erase() method should print “Erasing Circle”, “Erasing Triangle”, “Erasing Square” respectively.

Create objects of Circle, Triangle and Square in the following way and observe the polymorphic nature of the class by calling draw() and erase() method using each object.

Shape c=new Circle();

Shape t=new Triangle();

Shape s=new Square();

Wrapper Class

1. Write a java program which generates the minimum and maximum value for each of the Numeric Wrapper classes (Byte, Short, nteger, Long, Float, Double)

Sample Output:

Integer range:

min: -2147483648

max: 2147483647

Double range:

min: 4.9E-324

max: 1.7976931348623157E308

Long range:

min: -9223372036854775808

max: 9223372036854775807

Short range:

min: -32768

max: 32767

Byte range:

min: -128

max: 127

Float range:

min: 1.4E-45

max: 3.4028235E38

2.Accept a integer number as Command line argument from the program and when the program is executed print the binary, octal and hexadecimal equivalent of the given number.

Sample Output:

java Test 20

Given Number :20

Binary equivalent :10100

Octal equivalent :24

Hexadecimal equivalent :14

Refer Java Documentation and look for the appropriate Wrapper class method

3. Define a java class that accepts an integer(between 1 and 255) from the user and displays the String representation of the argument passed as an unsigned integer in base 2. The output displayed should contain 8 digits and should be padded with leading 0s(zeros), in case the returned String contains less than 8 characters.

For example, if the user enters the value 16, then the output should be

00010000

and if the user enters the value 100, the output should be

01100100

You are expected to use Integer class conversion method/s described in the PDF file. Use Scanner class to accept user inputs.(Hint : You may use String.format() method for the expected output)

4. Create an employee object and initialize its properties. Create a clone of this object and store it in a different object. Now change the properties of the first employee object. Print both the objects and observe the change.