Assignment -3 Method Overloading & Constructors

|  |  |
| --- | --- |
| class Cs  {  int p,q;  public Cs(){}  public Cs(int x, int y)  {  p=x;  q=y;  }  public int add(int i, int j)  {  return (i+j);  }  public int add(int i, int j, int k)  {  return (i+j+k);  }  public float add(float f1, float f2)  {  return (f1+f2);  }  public void printData()  {  System.out.print(“p = “+p);  System.out.println(“ q = “+q);  }  }  class Hari  {  public static void main(String args[])  {  int x=2, y=3, z=4;  Cs c=new Cs();  Cs c1=new Cs(x, z );  c1.printData();  float m=7.2F, n=5.2F;  int k=c.add(x,y);  int t=c.add(x,y,z);  float ft=c.add(m, n);  System.out.println(“k = “+k);  System.out.println(“t = “+t);  System.out.println(“ft = “+ft);  }  }  **Program explains the concept of method overloading and constructor overloading.** | import java.lang.\*;  class Comp  { float real,img;  public void getComp(float a,float b)  { real=a;img=b; }  public void print1()  {  System.out.println(real+"+"+img+"i");  }  public void duble()  { real=2\*real; img=2\*img; }  public void ktimes(int k)  { real=k\*real;img=k\*img; }  private float magsq()  { return real\*real+img\*img;}  public float magnitude()  { float t;  t=(float)Math.sqrt(this.magsq());  return(t);  }  public void add(Comp x)  { real=real+x.real;img=img+x.img; }  public void add(Comp x,Comp y)  {  real=x.real+y.real;img=x.img+y.img;  }  public Comp add3(Comp x)  { Comp c;float a,b;  c=new Comp();  a=real+x.real;  b=img+x.img;  c.getComp(a,b);  return(c);  }  public float getreal()  { return(real); }  private float getimaginary()  { return(img);}  }  class hari  {public static void main( String args[])  { Comp t,s,r,m;float j;  t=new Comp();s=new Comp();  r=new Comp();  t.getComp(4,2);t.duble();t.print1();  t.getComp(4,2);t.ktimes(10);  t.print1();  t.getComp(4,3);j=t.magnitude();  System.out.println(j);  s.getComp(2,7);t.getComp(4,2);  s.add(t);s.print1();  s.getComp(2,7);t.getComp(4,3);  r.add(s,t);r.print1();  s.getComp(2,7);t.getComp(4,23);  r=s.add3(t);r.print1();  System.out.println(s.getreal());  }  }  Since getreal( ) is public s.getreal( ) can be written in class hari. However s.getimaginary( ) can not be written.  s.add(t) means s+=t; r.add(s,t) means r=s+t; Two definitions show method overloading.  r=s.add3(t) means r=s+t; |

1. Write a JAVA program which contains a function square() such that square(3) returns 9, square(0.2) returns 0.04.
2. Write a JAVA program which contains a function cube() such that cube(3) returns 27, cube(0.2) returns 0.008.
3. Write a JAVA program which contains a function fun() such that fun(x) returns x2 and fun(x,y) returns x2 + y2. (where x and y are integers).
4. Write a JAVA program which contains a function fun() such that fun(x) returns x and fun(x,y) returns x + y and fun(x,y,z) returns x\*y\*z. (where x, y and z are integers).
5. Write a set of overloaded functions **min()** that returns the smaller of two numbers passed to them as arguments. Make versions for int and float.
6. Write a set of overloaded functions **power()** that returns the Xn where n is integer and X may be int and float.
7. Write a set of overloaded functions **max()** that returns the biggest of two numbers passed to them as arguments. Make versions for int and float.
8. Implement a=a\*b as a.mul(b), where a and b are objects of class Comp.
9. Implement a=b\*c as a=Comp.mul(b, c), where a, b and c are objects of class Comp.
10. Define Comp conjugate() e.g. conjugate(2+3i) is 2-3i
11. Implement a=b/c as a=b.divide(c), where a, b and c are objects of class Comp.

[Hint: a/b is defined as a\*b.conjugate()/b.magnitude2()]

1. Implement a=b/c as a=Comp.divide(b, c), where a, b and c are objects of class Comp.

[Hint: a/b is defined as a\*b.conjugate()/b.magnitude2()]