Esercizio 6

Realizzare una classe astratta per le Figure piane e due sottoclassi, la sottoclasse Quadrato e la sottoclasse Rettangolo.

Soluzione esercizio 6

```
public abstract class FiguraPiana2{

        private double base;

        public FiguraPiana2(double b){
        base=b;}

public abstract double Perimetro();
public abstract double Area();

public double getBase(){
return(base);}
}
```

Classe Quadrato che implementa Figura Piana

```
public class Quadrato2 extends FiguraPiana2{

public Quadrato2(double b){
    super(b);
    }

public double Perimetro(){
return (super.getBase()* 4);
}

public double Area(){
return (Math.pow(super.getBase(),2));
        }
}
```

Classe Rettangolo che implementa Figura Piana

```
public class Rettangolo2 extends FiguraPiana2{
 private double altezza;

public Rettangolo2(double b, double a){
super(b);
 altezza=a; }

public double Perimetro(){
return (super.getBase()* 2)+(altezza * 2);}

public double Area(){
return (super.getBase()* altezza);}

public double getAltezza(){
return(altezza);}}
```

Esercizio 7: Conto Bancario

Scrivere un programma che costruisca un conto bancario chiamato *conto1*, versi in esso $1000, prelevi da esso $500, prelevi altri $400 e infine visualizzi il saldo rimanente.
Il programma deve poi creare un altro conto bancario non vuoto chiamato *conto2*.
Su quest'ultimo conto deve essere poi applicato un interesse del 10%, a seguito del quale viene stampato il saldo.

Soluzione esercizio 7

```
public class BankAccount {
private double balance;

//si costruisce un conto bancario con saldo uguale a zero
public BankAccount()
    {  balance = 0; }

/* si costruisce un conto bancario con un saldo assegnato
initialBalance= il saldo iniziale */
 public BankAccount(double initialBalance)
{  balance = initialBalance; }

     // si versa denaro nel conto bancario,  amount= l'importo da versare
 public void deposit(double amount)
{  balance = balance + amount; }

//si preleva denaro dal conto bancario ,   amount= l'importo da prelevare
    public void withdraw(double amount)
    {  balance = balance - amount; }
```

```
    // s'ispeziona il valore del saldo attuale del conto bancario
public double getBalance()
    { return balance; }

   // si calcola un interesse sul conto, rate= il tasso d'interesse
    public void addInterest(double rate)
{      balance = balance + ((balance * rate) / 100); }

public static void main(String[] args)
{     BankAccount conto1 = new BankAccount();
   conto1.deposit(1000);
   conto1.withdraw(500);
   conto1.withdraw(400);
   System.out.println("Conto 1 : " +conto1.getBalance());
   BankAccount conto2 = new BankAccount(1000);
   conto2.addInterest(10); //Interessi al 10%
   System.out.println("Conto 2 : " + conto2.getBalance());
      }}
```

Esercizio 8: Conto Bancario con Password

Lo scopo del programma è riadattare il programma precedente (Esempio 7), introducendo il metodo boolean controlloPassword, che restituisce true se la password inserita corrisponde a quella del conto corrente cercato, e il metodo boolean controlloPrelievo, che restituisce true se la somma inserita è disponibile nel conto corrente.

Soluzione esercizio 8

```
import java.util.Scanner;
public class BankWithPassword  {
private double balance;
 private int password;

 /*si costruisce un conto con saldo uguale a zero e password di accesso
 numerica, pass=password numerica di accesso al conto*/
public BankWithPassword(int pass)
  {balance = 0;  password = pass;}

 /*si costruisce un conto con un saldo assegnato e password di accesso
initialBalance= il saldo iniziale  pass= password di accesso al conto*/
public BankWithPassword(double initialBalance, int pass)
  {balance = initialBalance;
  password = pass;}

/* si versa denaro nel conto bancario, amount= l'importo da versare*/
public void deposit(double amount)
  { balance = balance + amount;}
```

```
/* si preleva denaro dal conto bancario, amount= l'importo da prelevare*/
public void withdraw(double amount)
  { balance = balance - amount;}

 /* si applica un interesse sul conto, rate= il tasso d'interesse*/
 public void addInterest(double rate)
  { balance = balance + ((balance * rate) / 100);}

 /* ispeziona il valore del saldo attuale del conto bancario,  restituisce il saldo
attuale*/
 public double getBalance()
  { return balance; }

/* restituisce la password del conto*/
 public int getPassword()
 { return password;}

 /* verifica la validità della password immessa, pass= la password da verificare
 restituisce true se corretta, false se errata*/
 public boolean controlloPassword(int pass)
  { if (pass == password)   return true;
  return false;}
```

```
/* verifica che la somma da prelevare sia disponibile nel conto, amount = la somma da prelevare , restituisce true se disponibile, false se non disponibile*/
 public boolean controlloPrelievo(double amount)
  { if (amount <= balance)
   return true;
  return false;}

 public static void main(String[] args)
 {   //si inizializza la variabile che mi servirà per inserire i dati  dalla tastiera
  Scanner in=new Scanner(System.in);

  //si costruisce un primo conto con saldo iniziale pari a zero e  password definita dall'utente
  System.out.println("Inserire un numero che sarà password del conto 1");
  BankWithPassword conto1 = new BankWithPassword(in.nextInt());

  //stampa di verifica dei dati inseriti
  System.out.println("Il saldo del conto 1 è: " + conto1.getBalance()
   + "\nla password del conto 1 è: " + conto1.getPassword());
```

```
//si costruisce un secondo conto con saldo iniziale e   password definiti dall'utente
  System.out.println("Inserire un numero che sarà password del conto 2");
  int temp = in.nextInt();
  System.out.println("Inserire il saldo iniziale del conto 2");
  BankWithPassword conto2 = new BankWithPassword(in.nextDouble(),  temp);

//stampa di verifica dei dati inseriti
  System.out.println("Il saldo del conto 2 è: " + conto2.getBalance()
   + "\nla password del conto 2 è: " + conto2.getPassword());

  //si preleva una somma dal conto 2
  System.out.println("Prelievo dal conto 2:\nInserire la password del conto 2");
  if(conto2.controlloPassword(in.nextInt()))//controlla la password
 {  System.out.println("Inserire la somma da prelevare:");
    double amount = in.nextDouble();
    //controlla che la somma sia disponibile e in caso affermativo la preleva
    if (conto2.controlloPrelievo(amount))
     conto2.withdraw(amount);
    else
     System.out.println("Somma non disponibile nel  conto");  }
  else    System.out.println("Password errata. Impossibile effettuare il prelievo.");
```

```
\\si versa una somma sul conto 1
 System.out.println("Versamento sul conto 1:\nInserire la password del conto 1");
 if(conto1.controlloPassword(in.nextInt()))//controlla la password
 {
   System.out.println("Inserire la somma da versare:");
   conto1.deposit(in.nextDouble());
 }
 else
  System.out.println("Password errata. Impossibile effettuare il versamento.");
 //stampa un resoconto dei saldi
 System.out.println("Nuovi saldi:\nconto 1: " + conto1.getBalance() + "\nconto 2: "
+ conto2.getBalance());
 //applica l'interesse sui due conti e stampa i saldi aggiornati
 conto1.addInterest(3);
 conto2.addInterest(3);
 System.out.println("Nuovi saldi con interesse applicato del 3%:");
 System.out.println("conto 1: " + conto1.getBalance() + "\nconto 2: "
  + conto2.getBalance());
 }}
```