

CLASSI ASTRATTE

Una classe astratta è una classe che non può essere istanziata.

Non ci si può creare un'istanza della classe astratta ma i suoi attributi, i suoi metodi e i suoi costruttori sono tutti accessibili.

La classe astratta è sempre una superclasse di una classe derivata

Una classe principale contiene le funzionalità comuni di un insieme di classi figlie, ma la classe madre stessa è troppo astratta per essere usato da solo.

Classe Astratta:

Per dichiarare una classe astratta si usa la parola chiave **abstract**.

```
/* Nome File : Employee.java */
public abstract class Employee
{
    private String name;
    private String address;
    private int number;
    public Employee(String name, String address, int number)
    {
        System.out.println("Constructing an Employee");
        this.name = name;
        this.address = address;
        this.number = number;
    }
    public double computePay()
    {
        System.out.println("Inside Employee computePay");
        return 0.0;
    }
    public void mailCheck()
    {
        System.out.println("Mailing a check to " + this.name
            + " " + this.address);
    }
    public String toString()
    {
        return name + " " + address + " " + number;
    }
    public String getName()
    {
        return name;
    }
    public String getAddress()
    {
        return address;
    }
    public void setAddress(String newAddress)
    {
        address = newAddress;
    }
    public int getNumber()
    {
        return number;
    }
}
```

Si noti che non c'è nulla di diverso in questa classe Employee.

La classe è ora astratta, ma ha comunque tre campi, sette metodi, e un costruttore.

```
/* nome File : AbstractDemo.java */
public class AbstractDemo
{
    public static void main(String [] args)
    {
        /* la successive istruzione NON è ammessa */
        Employee e = new Employee("George W.", "Houston, TX", 43);

        System.out.println("\n Call mailCheck using Employee reference--");
        e.mailCheck();
    }
}
```

Quando si compila la classe si avrà il seguente errore:

Employee.java:46: Employee è astratta; non possono essere istanziati

```
    Employee e = new Employee ("George W.", "Houston, TX", 43);
```

^

1 errore

Estensione astratta Classe: uso corretto

Siamo in grado di estendere la classe Employee in modo normale, come segue:

```
/* nome File : Salary.java */
public class Salary extends Employee
{
    private double salary; // stipendio annuale
    public Salary(String name, String address, int number, double salary)
    {
        super(name, address, number);
        setSalary(salary);
    }
    public void mailCheck()
    {
        System.out.println("Within mailCheck of Salary class ");
        System.out.println("Mailing check to " + getName()
            + " with salary " + salary);
    }
    public double getSalary()
    {
        return salary;
    }
    public void setSalary(double newSalary)
    {
        if(newSalary >= 0.0)
        {
            salary = newSalary;
        }
    }
    public double computePay()
    {
        System.out.println("Computing salary pay for " + getName());
        return salary/52;
    }
}
```

Vediamo il main

```

/* Nome File : AbstractDemo.java */
public class AbstractDemo
{
    public static void main(String [] args)
    {
        Salary s = new Salary("Mohd Mohtashim", "Ambehta, UP", 3, 3600.00);
        Employee e = new Salary("John Adams", "Boston, MA", 2, 2400.00);

        System.out.println("Call mailCheck using Salary reference --");
        s.mailCheck();

        System.out.println("\n Call mailCheck using Employee reference--");
        e.mailCheck();
    }
}

```

Ciò dovrebbe produrre il seguente risultato:

Costruito un Dipendente

Costruito un Dipendente

Chiamata a MailCheck con riferimento Salary -in MailCheck di classe Salary

Controllo Mail per Mario Rossi con stipendio 3600.0

Chiamata a MailCheck con riferimento Salary -in MailCheck di classe Salary

Controllo Mail per Pino Bianchi con stipendio 3600.0

Metodi Astratti:

La parola chiave `abstract` è anche usato per dichiarare un metodo come astratto. Un metodo astratto è costituito da una firma del metodo, ma nessun corpo del metodo.

```

public abstract class Employee
{
    private String name;
    private String address;
    private int number;

    public abstract double computePay();

    //Remainder of class definition
}

```

Dichiarare un metodo astratto produce due risultati:

La classe deve essere dichiarata astratta. Se una classe contiene un metodo astratto, la classe deve essere anche lei astratta.

Ogni classe figlia deve definire in `Overriding` il metodo astratto o dichiararsi astratta.