

Esempio: Interfacce

Descrizione

- Si supponga di avere delle carte da gioco. Ogni carta rappresenta un **Personaggio** che può essere di tipo o **Umano** o **Mostro**.
- Un Personaggio di tipo **Umano** ha una forza fisica iniziale pari al numero 10 e può **combattere**, un Personaggio di tipo **Mostro**, invece, ha una forza fisica iniziale pari a 15 e può **azzannare**.
- I Personaggi del gioco sono Eroe, Vampiro e Licantropo.
 - **Eroe** è solo della categoria **Umano**,
 - **Vampiro** è solo della categoria **Mostro**,
 - **Licantropo** è di tipo **Mostro** nelle notti di luna piena, altrimenti di tipo **Umano**.
- In particolare, la forza fisica dei personaggi diminuisce di un valore pari a:
 - 3 per l'Eroe ad ogni combattimento
 - 2 per il Vampiro ad ogni azzanno
 - 2 per il Licantropo nelle notti di luna piena, 3 nelle altre.

Specifiche

- Si dovranno definire tutte le classi e le interfacce necessarie per realizzare il gioco
- Si dovrà inoltre definire una classe principale che:
 - Istanza tre oggetti: uno di tipo Eroe, uno di tipo Licantropo e uno di tipo Vampiro.
 - Fa combattere tre volte l'Eroe
 - Fa combattere una volta il Vampiro
 - Fa combattere il Licantropo due volte.
 - Stampa al termine la forza fisica rimasta a ciascun personaggio

Modelliamo il problema

- Cosa conviene usare: subclassing o interfacce?
- Cosa modelliamo con le classi e cosa con le interfacce?
- La scelta corretta è usare:
 - le **interfacce** per modellare i **comportamenti**
 - le **classi** per modellare le **entità concrete**

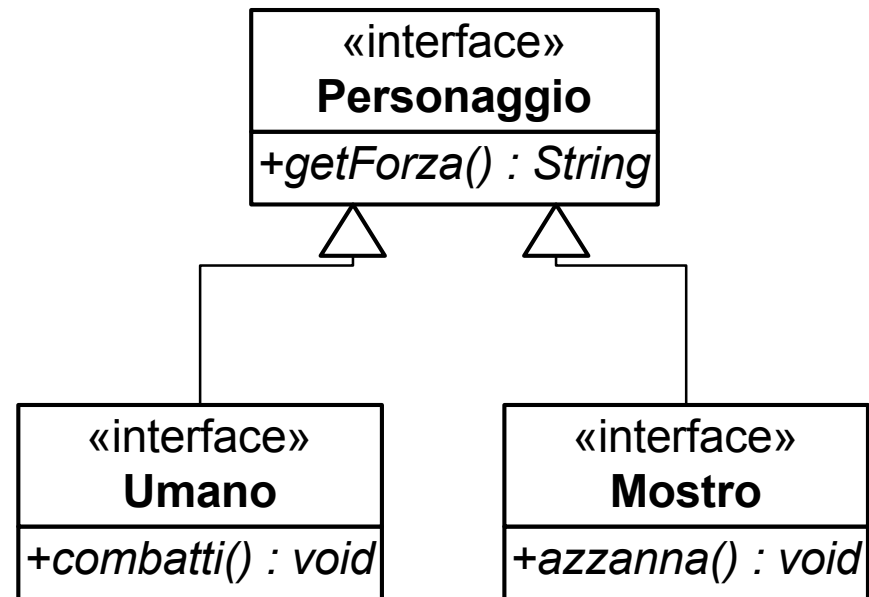
Nota: anche le interfacce sono organizzate in gerarchia

Interfacce: Personaggio, Umano e Mostro

```
public interface Personaggio
{
    public String getForza();
}
```

```
public interface Umano
    extends Personaggio
{
    public void combatti();
}
```

```
public interface Mostro
    extends Personaggio
{
    public void azzanna();
}
```



La classe Vampiro

```
public class Vampiro implements Mostro
{
    protected int forza;

    public Vampiro()
    {
        forza=15;
    }

    public void azzanna()
    {
        forza =forza-2;
    }

    public String getForza()
    {
        return "Forza rimanente come vampiro:"
        + forza;
    }
}
```

Vampiro
#forza : int
+Vampiro() +azzanna() : void +getForza() : String

La classe Eroe

```
public class Eroe implements Umano
{
    protected int forza;

    public Eroe()
    {
        forza= 10;
    }

    public void combatti()
    {
        forza=forza-3;
    }

    public String getForza ()
    {
        return "Forza rimanente come eroe:"
            + forza;
    }
}
```

Eroe
#forza : int
+Eroe() +combatti() : void +getForza() : String

La classe Licantropo

```
public class Licantropo implements Mostro, Umano
{
    private boolean isUomo;
    protected int forzaUmano, forzaMostro;
    public Licantropo(boolean luna)
    {
        isUomo = !luna;
        if (luna)
            {forzaMostro=15; forzaUmano=0; }
        else {forzaUmano=10; forzaMostro=0;}
    }
    public String getForza ()
    {
        return
            "Forza rimanente come umano:"+forzaUmano+
            "Forza rimanente come mostro"+forzaMostro;
    }
    public void azzanna()
    { if (!isUomo) forzaMostro=forzaMostro-2;}
    public void combatti ()
    { if (isUomo) forzaUmano=forzaUmano-3;}
}
```

Licantropo

-isUomo : bool
#forzaUmano : int
-forzaMostro : int

+Licantropo()
+combatti() : void
+azzanna() : void
+getForza() : String

Specifiche

- Si dovranno definire tutte le classi e le interfacce necessarie per realizzare il gioco
- Si dovrà inoltre definire una classe principale che:
 - Istanza tre oggetti: uno di tipo Eroe, uno di tipo Licantropo e uno di tipo Vampiro.
 - Fa combattere tre volte l'Eroe
 - Fa combattere una volta il Vampiro
 - Fa combattere il Licantropo due volte.
 - Stampa al termine la forza fisica rimasta a ciascun personaggio

Classe principale: Giochiamo

```
public class Giochiamo
{
    public static void main (String [] args)
    {
        Eroe e = new Eroe();
        Licantropo l = new Licantropo(true);
        Vampiro v = new Vampiro();
        for (int i =0; i<3;i++)
            e.combatti();
        v.azzanna();
        l.azzanna();
        l.azzanna();
        System.out.println(v.getForza());
        System.out.println(l.getForza());
        System.out.println(e.getForza());
    }
}
```

Giochiamo
<u>+main(in args : String[]) : void</u>

Il diagramma delle classi

