



UNIVERSITÀ DEGLI STUDI DI SALERNO
DIPARTIMENTO DI INFORMATICA
DIPARTIMENTO DI ECCELLENZA

Università di degli Studi di Salerno

Dipartimento di Informatica

Programmazione ad Oggetti

a.a. 2023-2024

Packages

Docente: Prof. Massimo Ficco

E-mail: *mficco@unisa.it*

I packages

Dobbiamo evitare confusioni tra classi con lo stesso nome

Organizzare il codice

Consentire lo sviluppo ed il riutilizzo di librerie



Definizione di un package

Le Classi Java sono raggruppate nei “package”.

- I package sono strutturati in uno schema ad albero:
aaa.bbb.ccc.MyClass
- Il nome di un package può essere composto da più identificatori separati da “.”:
roberto.utilities.internet
- Senza definire uno specifico package una classe verrà inserito nel default package.
- Solo all’inizio del file java si può specificare l’appartenenza ad un package:
package mypackage;
public class MyClass {
// ...



Packages

Più classi possono essere riunite in un **package**, dotato di nome

Il nome di una classe (o di una interfaccia) va sempre qualificato con il nome del package a cui appartiene

Es: java.swing.JOptionPane;

Es: java.util.Date;

Può essere omesso:

- quando viene usato all'interno dello stesso package
- utilizzando lo statement **import**



Esempio

```
// Stampa la data corrente
class prova{
    public static void main(String args[])
    {
        System.out.println(new java.util.Date());
    }
}
```



Esempio 2

```
Import java.util.Date; //importa solo la classe Data
//Import java.util.*;   //importa tutto il package
class prova{
    public static void main(String args[])
    {
        System.println(new Date());
    }
}
```



Creare un proprio package

Keyword **package**:

- Prima riga del file
- Deve apparire una sola volta

Relazione tra package e cartelle

Per evitare che package di produttori diversi abbiano lo stesso nome, si suggerisce di far iniziare il nome del package con il dominio Internet del produttore (invertito, e con il nome di più alto livello tutto maiuscolo):

www.unimi2.dsi.roberto.utilities.internet



Criteri di raggruppamento

Package access è un criterio per raggruppare classi:

- Conviene inserire nello stesso package classi correlate così che queste possano facilmente interagire
- Probabilmente ciò ci invita ad escludere classi che non vogliamo dipendano dalle prime



Un esempio

```
package geometria;
```

```
public class Triangolo{  
    float base =0;  
    float altezza=0;
```

```
    public Triangolo(float b, float h){  
        base=b;  
        altezza=h;  
    }
```

```
    public float area() {  
        return (base*altezza)/2;  
    }  
}
```



Un esempio

```
package geometria;
```

```
public class Quadrato{  
    float lato =0;
```

```
    public Quadrato(float l){  
        lato=l;  
    }
```

```
    public float area() {return l*l;}  
}
```



Un esempio

```
import geometria.*;

public class Test{
    public static void main(String args[]){
        Triangolo t=new Triangolo(5,3);
        Quadrato q=new Quadrato(4);
        System.out.println(t.area());
        System.out.println(q.area());
    }
}
```



Creare un proprio package

- Specificare il package che corrisponderà al percorso del filesystem a partire dal **CLASSPATH**

