

Classi e oggetti: *parte II*

Imparare a **realizzare** semplici classi di tipo *entity* nel linguaggio Java

Creare classi *Entity*

- Realizzazione della classe GradeBook
 - IDE
 - Costruttore
 - Riferimento *this*
 - Controllo dell'accesso ai membri di una classe
 - Metodi set/get
 - Passaggio riferimenti verso metodi

Progettazione “GradeBook”: problemi

- **Attributi:**

- `grades: int[]`

sequenza di interi → Array prossime lezioni

- **Metodi:**

- `getAverage()` (calcolo della media dei voti)

- `getMaximum()` (calcolo del massimo dei voti)

metodi per l’elaborazione dei voti:

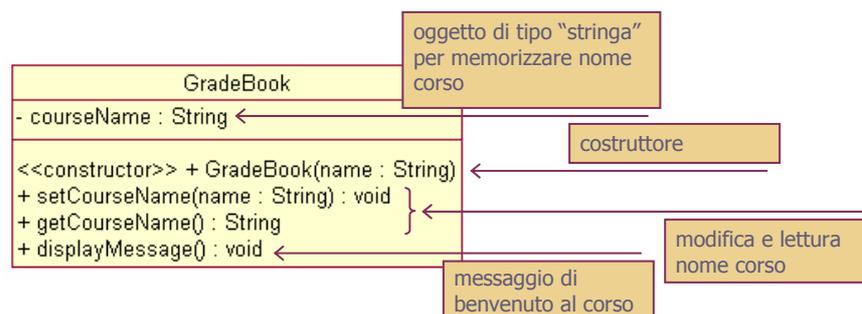
- progetto di algoritmi

- costrutti del linguaggio java (cicli e decisioni)

- uso di oggetti di tipi primitivi nei metodi

prossime lezioni

“GradeBook”: progetto *parziale*



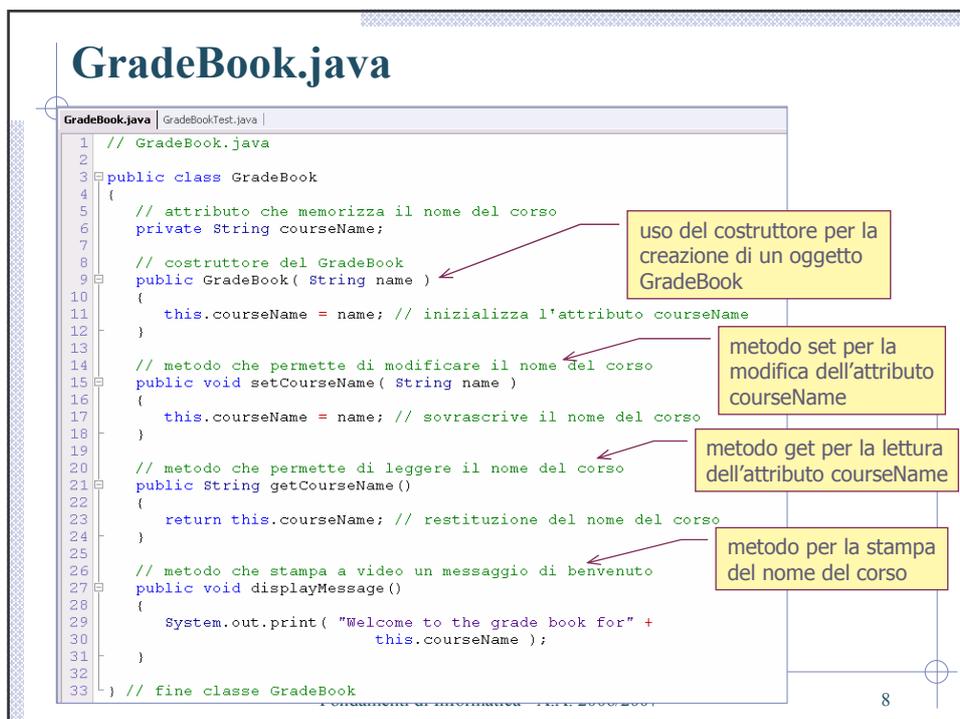
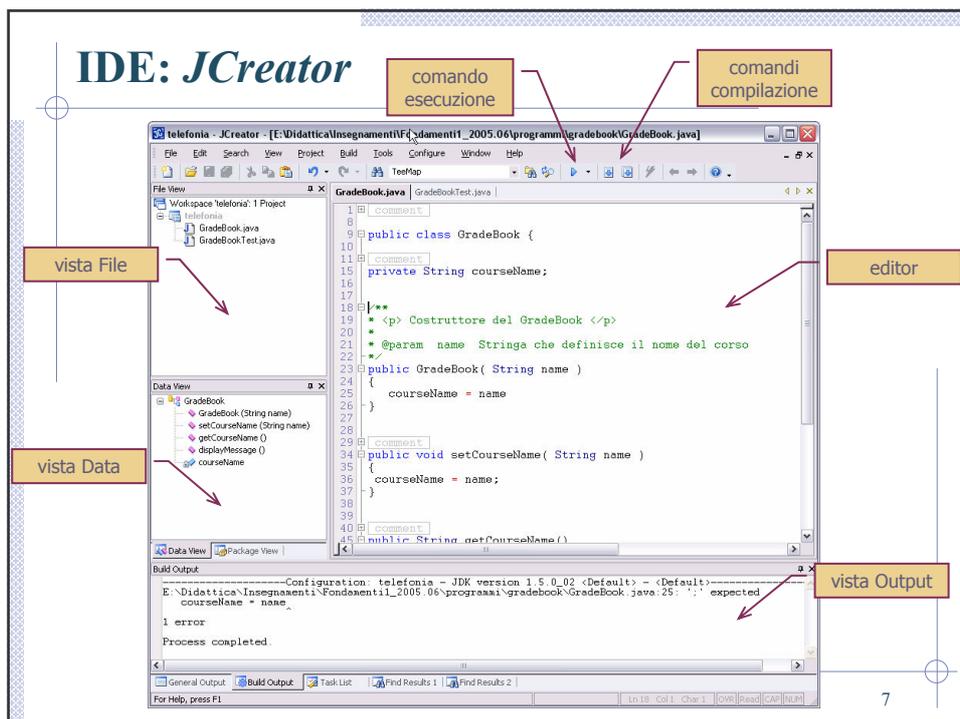
- Primo esempio (minimale) di progetto di una classe di tipo Entity
- Analisi di problematiche di base

Implementazione GradeBook

- Creazione file:
 - `GradeBook.java`
 - `GradeBookTest.java`
- Esecuzione:
 - compilazione (`javac *.java`)
 - esecuzione (`java GradeBookTest`)
 - errori "compile-time" e "run-time"

IDE

- Esistono ambienti di sviluppo integrati (Integrated Development Environment, IDE)
- Mettono a disposizione strumenti per lo sviluppo di programmi e la loro messa a punto
 - editor
 - compilatore e/o interprete
 - debugger



GradeBookTest.java

```
1 // GradeBookTest.java
2
3 public class GradeBookTest
4 {
5     // metodo principale (main), inizia l'esecuzione del
6     public static void main( String args[] )
7     {
8         // crea un oggetto GradeBook
9         GradeBook gb = new GradeBook( "Fondamenti di Informatica" );
10
11         // visualizza il valore di courseName dell'oggetto gb
12         System.out.println( "Il nome del corso di gradeBook e': " +
13                             gb.getCourseName() );
14
15         // modifica il nome del corso dell'oggetto gb
16         gb.setCourseName( "Geometria" );
17
18         // visualizza il valore di courseName dell'oggetto gb
19         System.out.println( "Il nome del corso di gradeBook e': " +
20                             gb.getCourseName() );
21
22         // visualizza il messaggio
23         gb.displayMessage();
24     } // end main
25 } // fine classe GradeBookTest
```

uso del costruttore per la creazione di un oggetto GradeBook

passaggio di un parametro al metodo setCourseName()

```
C:\PROGRA-1\XINOS-1\UCREAT-1\GE2001.exe
Il nome del corso di gradeBook e': Fondamenti di Informatica
Il nome del corso di gradeBook e': Geometria
Welcome to the grade book for Geometria
Press any key to continue...
```

Riferimento *this*

- Nell'implementazione delle classi si usa la parola chiave **this** per riferirsi implicitamente all'oggetto che in un certo momento sta eseguendo un certo metodo

dal metodo main di GradeBookTest:

```
String s1 = new String("Mat"); // (1)
String s2 = new String("Geo"); // (2)

GradeBook gb1 = new GradeBook( s1 ); // (3)
GradeBook gb2 = new GradeBook( s2 ); // (4)
```

il costruttore di GradeBook:

```
public GradeBook( String name )
{
    this.courseName = name;
}
```

- nell'esecuzione dell'istruzione (3) **this** è un riferimento all'oggetto **gb1**
- nell'esecuzione dell'istruzione (4) **this** è un riferimento all'oggetto **gb2**

Public e private

- Le parole chiave `public` e `private` servono come modificatori d'accesso per attributi e metodi.

implement. A della classe `GradeBook`:

```
public class GradeBook {  
    private String courseName;  
    ...  
}
```

implement. B della classe `GradeBook`:

```
public class GradeBook {  
    public String courseName;  
    ...  
}
```

dal metodo `main` di `GradeBookTest`:

```
public static void main( String[] args ) {  
    GradeBook gb = new GradeBook( "Geometria" );  
    gb.courseName = "Analisi Matematica";  
    ...  
}
```

Domanda: qual è il risultato della compilazione nei casi (A) e (B) ?

Riflessione

- Domanda d'esame:
 - la classe `GradeBook` prevede entrambi i metodi `set` e `get` per l'attributo `courseName`
 - con questi metodi è possibile leggere e modificare lo stato dell'attributo `courseName`
- Dire se è vera o falsa la seguente affermazione:
 - sono totalmente equivalenti le seguenti strategie:
 - definire un attributo di tipo `public`
 - definire un attributo `private` ed includere entrambi i metodi `set` e `get` per l'attributo in questione

Tipi di dato primitivi e non

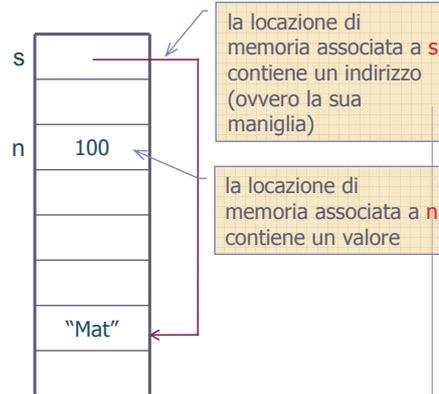
Primitivi

- **boolean**, **char**, numeri interi, numeri reali

Non primitivi (Riferimenti)

- oggetti
- **null** (valore speciale)

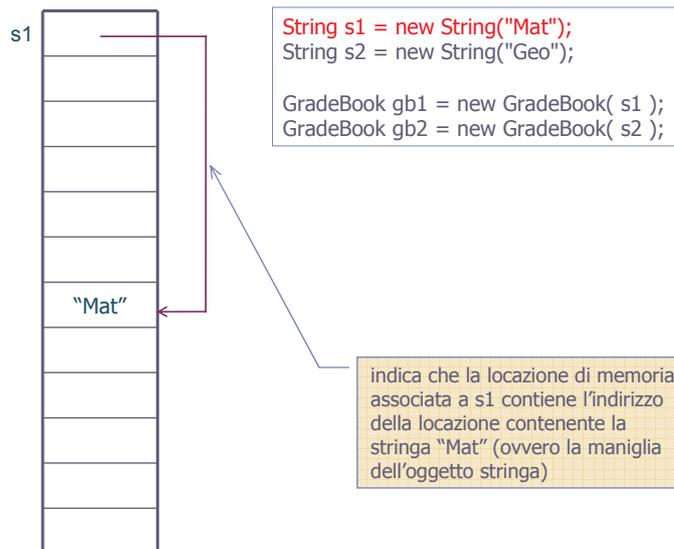
```
String s = new String("Mat");  
int n = 100;
```



Memoria e parametri (1)

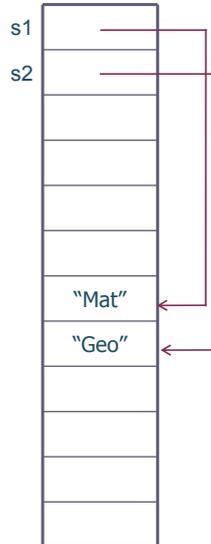
(1)

```
String s1 = new String("Mat");  
String s2 = new String("Geo");  
  
GradeBook gb1 = new GradeBook( s1 );  
GradeBook gb2 = new GradeBook( s2 );
```



Memoria e parametri

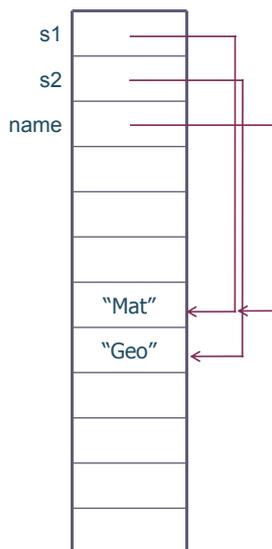
(2)



```
String s1 = new String("Mat");  
String s2 = new String("Geo");  
  
GradeBook gb1 = new GradeBook( s1 );  
GradeBook gb2 = new GradeBook( s2 );
```

Memoria e parametri

(3)



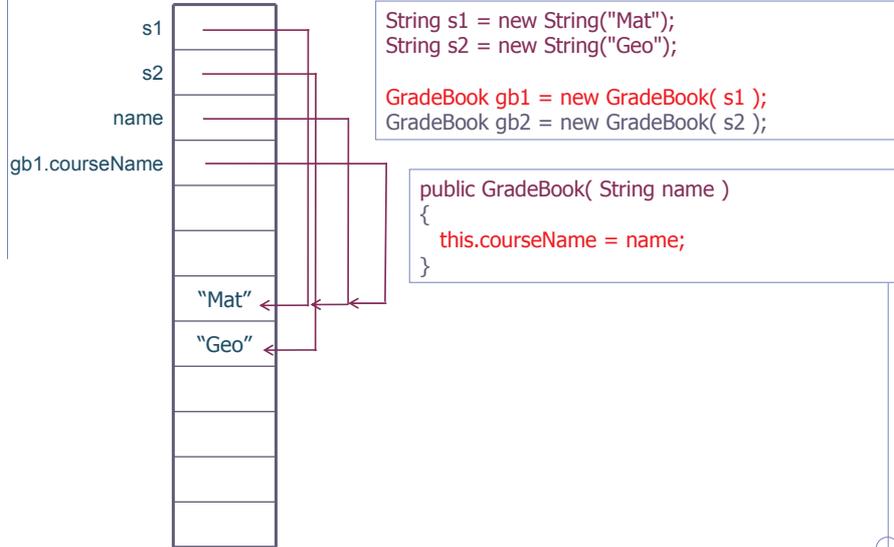
```
String s1 = new String("Mat");  
String s2 = new String("Geo");  
  
GradeBook gb1 = new GradeBook( s1 );  
GradeBook gb2 = new GradeBook( s2 );
```

```
public GradeBook( String name )  
{  
    this.courseName = name;  
}
```

name = s1

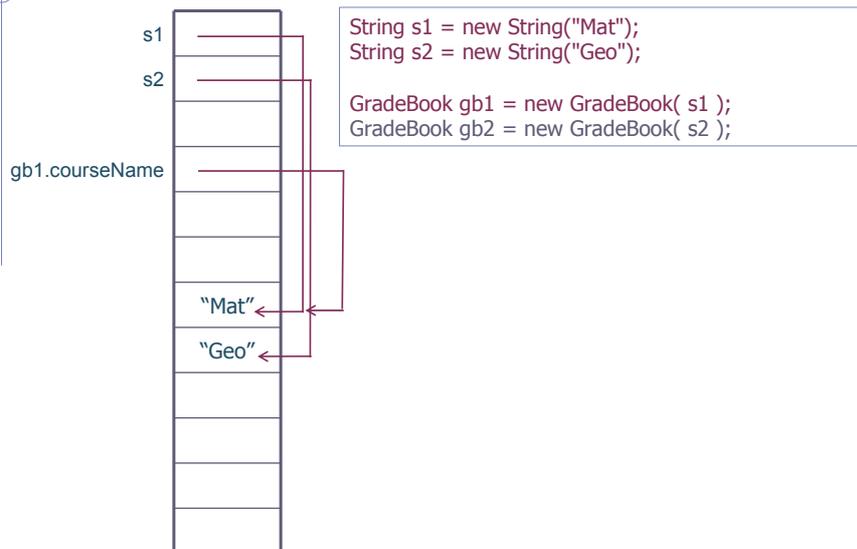
Memoria e parametri

(4)



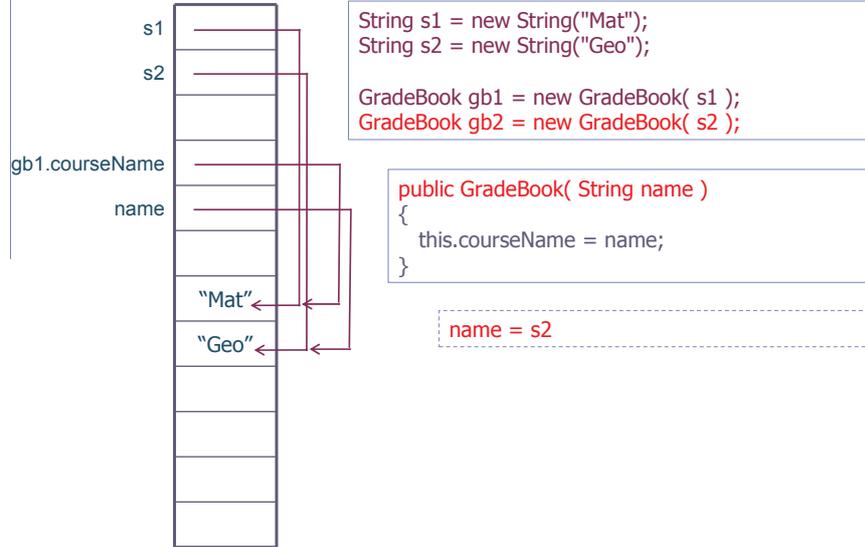
Memoria e parametri

(5)



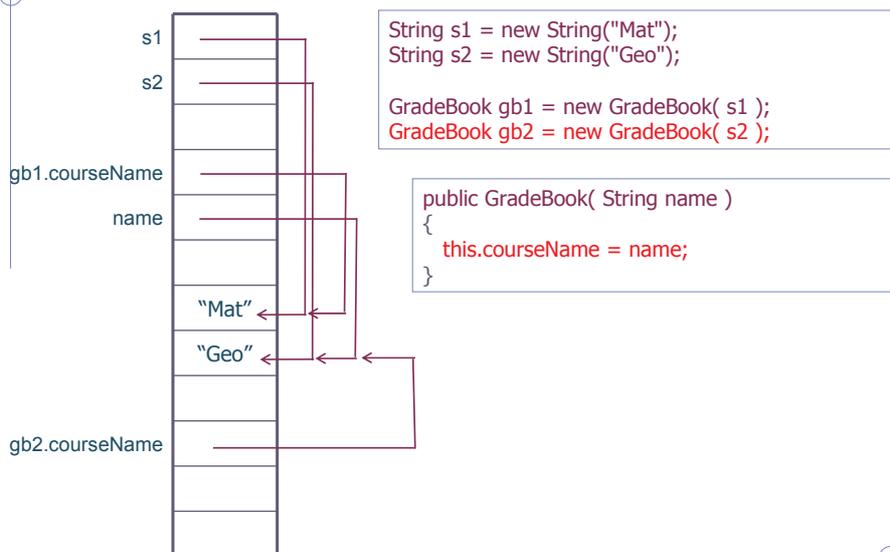
Memoria e parametri

(6)



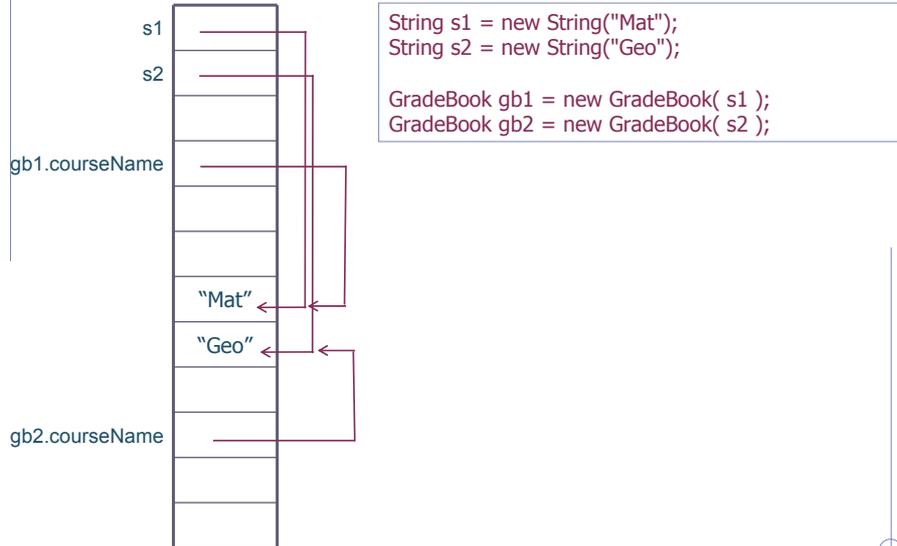
Memoria e parametri

(7)



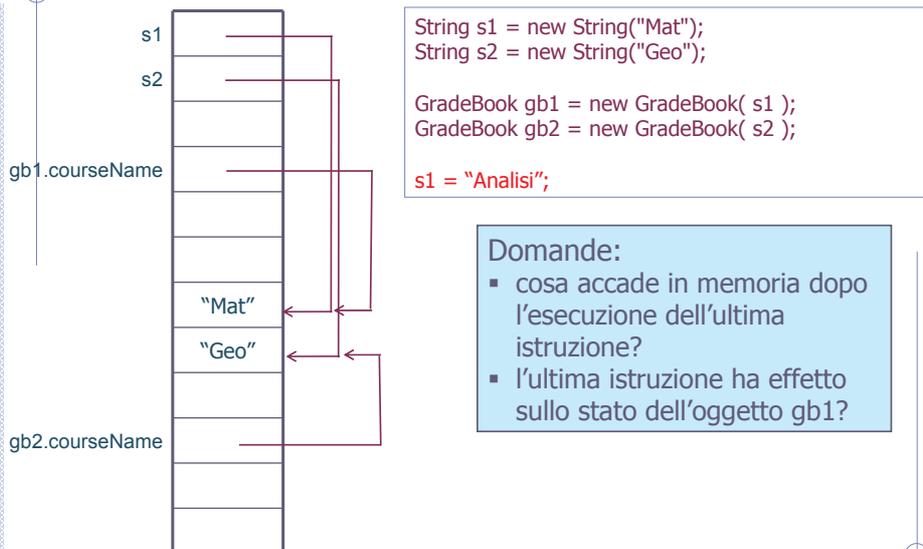
Memoria e parametri

(8)



Memoria e parametri

(9)



Domande:

- cosa accade in memoria dopo l'esecuzione dell'ultima istruzione?
- l'ultima istruzione ha effetto sullo stato dell'oggetto gb1?

Esercizio 2

- Modificare la classe GradeBook come segue
 - Aggiungere una variabile di istanza di tipo string che rappresenti il nome dell'insegnante del corso
 - Fornire i metodi set e get per tale variabile
 - Modificare il costruttore in modo da creare oggetti della nuova classe GradeBook
 - Modificare il metodo displayMessage in modo che, dopo aver visualizzato il messaggio di benvenuto, scriva "Questo è il corso tenuto dal Professor: " seguito dal nome dell'insegnante
- Utilizzare la nuova classe GradeBook all'interno di una applicazione di test per verificare le nuove funzionalità

Prossima lezione

- Definizione di **algoritmo**
- La nozione di **controllo**
- Strutture di **controllo** in Java
 - Struttura **sequenziale**
 - Strutture **di selezione** (if, if-else)
 - Strutture **di iterazione** (while)
- Esempi d'uso