

Classe 4CI

A.S. 2017/18

Classe 4C INFO

Sintesi del Programma svolto durante l'anno scolastico 2017/18

(il dettaglio nel documento Piano consuntivo)

1) Introduzione alla metodologia object oriented

- 1.1) Paradigmi di programmazione
- 1.2) Dalla programmazione strutturata alla programmazione ad oggetti

2) Object Oriented Programming

- 2.1) Concetto di classe
- 2.2) Concetto di attributo
- 2.3) Concetto di metodo
- 2.4) Concetto di oggetto (istanza di classe)
- 2.5) Forme simboliche di rappresentazione di classi e oggetti
 - 2.5.1) UML: Unified Modeling Language – Linguaggio di modellazione
 - 2.5.1.1) Class diagram
 - 2.5.1.2) Use Case diagram
 - 2.5.1.3) Schema di rappresentazione grafica di oggetti
- 2.6) Associazioni tra classi
- 2.7) Concetto di interazione tra oggetti /scambio messaggi/invocazione di metodi
- 2.8) Information hiding (Incapsulamento)
 - 2.8.1) modificatori public/private/protected
 - 2.8.2) modificatori Static e Final
 - 2.8.3) metodi get() e set()
 - 2.8.4) Parametri di un metodo
 - 2.8.4.1) Metodi che ritornano void
 - 2.8.4.2) passaggio parametri per valore e per riferimento
 - 2.8.4.3) passaggio di oggetti come parametri

2.9) Costruttori

2.9.1)Costruttore di default

2.9.2)Costruttore di copia

2.10)Ereditarietà

2.10.1) Superclasse e sottoclasse

2.11)Concetto di Polimorfismo

2.12)Overriding dei metodi

2.13)Overloading dei metodi

2.14)Classi wrapper

2.15)Classi astratte ed interfacce

2.16)Le stringhe

2.16.1) Stringhe come oggetti immutabili

2.16.1) Metodi di utilità sulle stringhe

2.17) Strutture dati statiche: Gli array

3)Il linguaggio Java

3.1)Costrutti generali del linguaggio

3.2)Il JDK (Java Development Kit)

3.3)JRE (Java Runtime Environment)

3.4)La Java Virtual Machine (JVM)

3.5)Il Garbage Collector

3.6)Utilizzo di NetBeans come ambiente di sviluppo

3.7)Primi programmi Java per sperimentare i tool di sviluppo ed esecuzione

3.7.1) java compiler (comando javac) e la generazione del file .class

3.7.2) esecuzione di classi (comando java)

3.7.3) Il metodo Main()

3.8) Esercitazioni Java per sperimentare i concetti appresi nelle lezioni teoriche

3.9)Gerarchia delle classi di java (Da Object)

3.10) La documentazione automatica (Javadoc)

3.11)Array in java

3.11.1) Array di oggetti

3.12)Eccezioni in java

3.12.1)La gerarchia delle classi exception

3.13)Gestione dell'I/O (Input ed Output)

3.14)Gestione dell'I/O da file (vedi punto 4)

3.15)Serializzazione e persistenza degli oggetti su file (vedi punto 4)

3.16)Array dinamici: vector

3.17)Strutture dati (liste, pile, code,alberi)

3.18)Introduzione alla GUI (Graphic User Interface)

3.18.1) GUI(Graphic User Interface):Utilizzo delle librerie Awt e Swing

4) Introduzione alla gestione dei dati

4.1) Dati su archivio

4.1.1) Lettura/scrittura su file testo

4.1.2) Lettura/scrittura di oggetti su file strutturati (Serializzazione)

4.1.4) Il formato json e il formato XML

Laboratorio:

Durante le lezioni di laboratorio (3 ore settimanali), sono stati assegnati esercizi da svolgere individualmente e a gruppi, con difficoltà mano mano crescenti, che hanno permesso agli allievi di mettere in pratica ed approfondire i concetti appresi durante le lezioni teoriche.

Si è partiti nel primo periodo con l'utilizzo di java in modalità comando, senza un vero e proprio IDE, per sperimentare l'editing, la sintassi del linguaggio, la consapevolezza delle fasi di compilazione ed esecuzione delle classi java.

Inizialmente si trattava di gestire classi semplici volte alla simulazione di oggetti a scelta dai singoli allievi, per implementare inizialmente semplici azioni quali metodi get() e set(), SiPresenta() etc. Successivamente si è passati a svolgere esercizi volti al trattamento di costrutti più complessi quali stringhe, strutture dati inizialmente statiche (array di

oggetti) e poi dinamiche (Vector). Successivamente si è sperimentato l'uso dell'ereditarietà e del polimorfismo, quindi di classi astratte ed interfacce.

Nel passaggio all'uso della GUI (Graphic User Interface), si sono proposti agli allievi degli esercizi inizialmente semplici (utilizzo dei componenti e contenitori, disegni di figure geometriche) che successivamente sono stati arricchiti con elementi di grafica più complessa (gestione parametrica delle figure geometriche, gestione di un semplice editor di linee (implementazione grafica di una pila di comandi generati dal mouse, al fine di simulare un editor grafico come JOSM, utilizzato nel progetto di alternanza scuola lavoro OSM)..

Infine, il progetto robocode , proposto a dicembre, è stato utile quale stimolo per l'uso del linguaggio java volto alla realizzazione di codice per lo sviluppo di un robot utilizzato per un gioco.

Esercitazioni per le vacanze

Si propongono agli allievi alcune attività guidate per un ripasso generale degli argomenti trattati in quarta ed un approfondimento di alcuni argomenti utili per il proseguimento in quinta delle attività didattiche e di alternanza.

Le esercitazioni riguardano un primo approccio con il linguaggio java al concetto di database, un primo approccio al concetto di applicazione client/server con uso di Servlet e un approccio alla gestione di dati strutturati con formati di interscambio come json.

Seguono le proposte per le esercitazioni:

- 1) Rendiconto dell'esperienza di Stage effettuata dall'allievo**
Scrivere un documento di sintesi che spieghi l'esperienza di Stage (Minimo: 2 pagine)
- 2) Iniziare a ragionare al progetto da presentare all'esame di stato. Potrebbe essere un proprio progetto e/o un progetto di Alternanza scuola lavoro da presentare con particolare dettaglio.**

Iniziare quindi a preparare della documentazione relativa.

Non dimenticate di tenere un archivio con tutta la documentazione delle attività di alternanza che avete svolto fin dalla classe terza. Raccogliete il tutto e magari mettete tutto in un sito Web (ad es uno dei CMS che avete prodotto durante l'anno scolastico con la materia TPSI)

- 3) **Breve sintesi di un libro che avete letto durante l'anno e che riguardi il mondo della tecnologia informatica in genere**

- 4) **Riprendere sotto mano il materiale relativo alle esercitazioni svolte durante l'anno scolastico e ripassare, modificare e migliorare i programmi java svolti, curare la documentazione e prendere spunto da essi per rivedere i concetti teorici appresi durante l'anno.**

- 5) **Esercitazione con Java e MySQL: Entrare in area didattica del registro elettronico (compiti vacanze.zip: estrarre java e mysql)**

si tratta di due classi da provare:

-test.java

- scorri.java

Si tratta di creare un piccolo database con il tool MySQL.

1. Dapprima si deve Installare XAMPP sul proprio PC.
2. Attivare XAMPP, aprire MYSQL Admin e creare una semplice tabella denominata "Elenco" con due campi: nome Varchar(30) e cognome Varchar(30)
3. Inserire almeno 3 o 4 righe contenente dei dati dell'elenco.
4. Installare il driver jdbc mysql-connector-java-5.1.19-bin.jar e collocarlo il driver sotto la cartella relativa a jre\lib\ext:
es sotto: C:\Programmi\Java\jdk1.7.0_05\jre\lib\ext

5. Compilare e provare i due sorgenti test.java e scorri.java

Cercare di comprendere il funzionamento di MySQL e di java per accedere ad esso.

6. Scrivere un documento che spieghi l'esperienza fatta

- 6) **Esercitazione con Servlet java: Entrare in area didattica (compitivacanze.zip: Servlet)**

Provare a creare una servlet in java con la documentazione e il codice allegato.

Cercare di comprendere il funzionamento delle Servlet e scrivere un documento che spieghi l'esperienza fatta

7) Esercitazione con java e json:

Da un sorgente java, provare a leggere e scrivere un file json contenente i dati di temperatura, umidità, con data e ora di rilevamento.

Cercare di comprendere il formato json e scrivere un documento che spieghi l'esperienza fatta.