

Esempi di domande

Questo documento riporta, in ordine sparso, *esempi* di domande che potrebbero essere assegnate nella prova scritta o nel colloquio.

L'elenco non è da considerarsi assolutamente esaustivo!

Dovrebbe essere inutile precisare che chi si presenta all'esame deve essere in grado di rispondere con precisione e sicurezza a tutte le domande elencate.

Controllate eventuali aggiornamenti.

Si ricorda che nella prova scritta saranno assegnati anche alcuni esercizi, sullo stile di quelli proposti nel libro di testo e di quelli assegnati nelle prove scritte e nei compiti degli anni precedenti, disponibili sul sito del corso.

- Descrivete cosa è e a cosa serve un *compilatore*.
- Spiegate cosa si intende per *interprete*.
- Spiegate cosa è il *bytecode*.
- Presentate sinteticamente il tipo primitivo `boolean`, spiegando in particolare cosa si intende per *lazy evaluation*.
- Promozioni e forzature tra tipi primitivi.
- I significati della parola riservata `super`.
- I significati della parola riservata `this`.
- Presentate i vari significati dell'operatore `+` e spiegate in che momento della vita di un programma e in che modo viene attribuito a tale operatore il significato corretto.
- Spiegate cosa si intende per *overloading* dei metodi e come viene risolto nel linguaggio Java.
- Spiegate cosa si intende per *overriding* dei metodi e come viene risolto nel linguaggio Java.
- Cosa sono le classi involucro e a cosa servono?
- I tipi primitivi per la rappresentazione dei numeri interi.
- I tipi primitivi per la rappresentazione dei numeri in virgola mobile.
- Cosa si intende per classe astratta?
- Illustrate brevemente il costrutto `try/catch`.
- Tipi primitivi e tipi riferimento.

- Il significato di `throws` nell'intestazione di un metodo.
- Spiegate cosa si intende per *campo* e per *campo statico*.
- Spiegate cosa si intende per *metodo* e per *metodo statico*.
- Spiegate cosa si intende per *superclasse* e per *sottoclasse*. Fornite *poi* un esempio.
- Discutete il concetto di *metodo astratto*.
- Eccezioni controllate e non controllate.
- Presentate i diversi significati del modificatore `final`.
- Spiegate cosa si intende per array, indicando le caratteristiche fondamentali degli array nel linguaggio Java.
- Spiegate a cosa serve e come si utilizza l'operatore `instanceof`, fornendo alcuni esempi.
- Spiegate come è organizzata la memoria della macchina durante l'esecuzione.
- Spiegate che cosa è un *record di attivazione* e come è organizzato internamente.
- Spiegate a cosa serve e come è gestita l'area di memoria denominata *stack*.
- Spiegate a cosa serve e come è gestita l'area di memoria denominata *heap*.
- Spiegate la differenza tra selezione e iterazione, mostrando alcuni esempi concreti.
- Spiegate cosa si intende per *polimorfismo*. Fornite anche un esempio.
- La classe `Object` e la gerarchia delle classi.
- Il tipo `Object` e la gerarchia dei tipi.
- Discutete il legame tra i costruttori di una classe e quelli della superclasse.
- Spiegate cosa si intende per *eccezione* e illustrate cosa avviene nel caso durante l'esecuzione di un programma si verifichi un'eccezione che non viene intercettata.
- Spiegate cosa si intende per *eccezione* e come sia possibile *intercettare* le eccezioni.
- Spiegate cosa si intende per *eccezione* e come sia possibile *sollevare* le eccezioni in un metodo o in un costruttore.
- Discutete i modificatori `private` e `public` indicando il loro significato, dove possono essere utilizzati, e come è opportuno utilizzarli.
- Spiegate il concetto di *supertipo* e *sottotipo* facendo riferimento esplicitamente sia alle classi, sia alle interfacce.
- Descrivete brevemente cosa si può trovare nel corpo di una classe, specificandone la funzione.
- Discutete la differenza tra *variabili di istanza* e *variabili di classe*.
- Fornite lo schema generale di un ciclo `for`, spiegate il funzionamento e, infine, presentate e discutete un esempio.
- Indicate con precisione lo scopo di ciascuna parola che appare in questa intestazione di metodo `main`, da cui è possibile avviare l'esecuzione di un programma:

```
public static void main (String[] args)
```

-
- Presentate gli operatori di *congiunzione* e *disgiunzione*, descrivendone il significato e il loro uso nel linguaggio Java.
 - Descrivete le *strutture di controllo fondamentali* della programmazione strutturata. Indicate poi i corrispondenti costrutti nel linguaggio Java.
 - Descrivete il concetto di variabile e di tipo. Presentate poi alcune caratteristiche fondamentali dei tipi nel linguaggio Java.
 - Descrivete sinteticamente gli 8 tipi primitivi del linguaggio Java.
 - Fornite lo schema dei cicli con controllo in testa e con controllo in coda nel linguaggio Java, evidenziandone caratteristiche e differenze.
 - Promozioni e forzature tra tipi riferimento.
 - Spiegate cosa si intende per *promozione implicita* nel caso di tipi primitivi. Fornite anche qualche esempio.
 - Spiegate che cos'è la *Java Virtual Machine*.
 - Spiegate quali possono essere gli effetti *in esecuzione* di operazioni di *forzatura (cast)* nel caso di tipi primitivi e nel caso di tipi riferimento.