

## Package java.lang

Il package java.lang è uno dei package fondamentali in Java, e viene automaticamente importato in ogni programma Java. Contiene le classi di base per la gestione di stringhe, numeri, oggetti, thread e altre operazioni di sistema. Qui sotto sono elencate le classi principali di java.lang con una breve descrizione di ognuna.

---

### 1. Object

- **Descrizione:** È la superclasse di tutte le classi in Java. Ogni classe in Java eredita da Object, quindi contiene metodi base come equals(), hashCode(), toString(), clone() e finalize().
  - **Principali Metodi:**
    - equals(Object obj): Confronta due oggetti per verificarne l'uguaglianza.
    - hashCode(): Restituisce un valore hash dell'oggetto.
    - toString(): Restituisce una rappresentazione in stringa dell'oggetto.
- 

### 2. String

- **Descrizione:** Rappresenta una sequenza immutabile di caratteri. La classe String è largamente utilizzata in Java per la manipolazione di stringhe.
  - **Principali Metodi:**
    - length(): Restituisce la lunghezza della stringa.
    - charAt(int index): Restituisce il carattere in una posizione specifica.
    - substring(int beginIndex, int endIndex): Estrae una sottostringa.
    - concat(String str): Concatena due stringhe.
- 

### 3. StringBuilder e StringBuffer

- **Descrizione:** Classi per la manipolazione di stringhe mutevoli. StringBuilder non è thread-safe, mentre StringBuffer è thread-safe.
- **Principali Metodi:**
  - append(String str): Aggiunge una stringa alla fine.
  - insert(int offset, String str): Inserisce una stringa a una data posizione.
  - delete(int start, int end): Elimina caratteri in un intervallo.

---

#### 4. Math

- **Descrizione:** Fornisce metodi per operazioni matematiche come funzioni trigonometriche, logaritmi, esponenziali e altro.
- **Principali Metodi:**
  - **abs(double a):** Restituisce il valore assoluto.
  - **sqrt(double a):** Restituisce la radice quadrata.
  - **pow(double a, double b):** Calcola la potenza.
  - **random():** Restituisce un numero pseudo-casuale.

---

#### 5. Integer, Double, Float, Long, Byte, Short e Boolean

- **Descrizione:** Classi wrapper per i tipi primitivi (rispettivamente int, double, float, long, byte, short e boolean). Permettono di gestire i tipi primitivi come oggetti.
- **Principali Metodi:**
  - **parseInt(String s):** Converte una stringa in un intero (es. per Integer).
  - **valueOf(String s):** Converte una stringa nel valore primitivo corrispondente.

---

#### 6. System

- **Descrizione:** Fornisce l'accesso a funzionalità di sistema, come input/output standard e variabili di ambiente.
- **Principali Metodi:**
  - **currentTimeMillis():** Restituisce l'ora corrente in millisecondi.
  - **exit(int status):** Termina l'applicazione.
  - **gc():** Richiama il garbage collector.

---

#### 7. Thread

- **Descrizione:** Rappresenta un thread di esecuzione. Java permette di creare programmi multi-threaded per eseguire operazioni parallele.
- **Principali Metodi:**

- **start():** Avvia l'esecuzione del thread.
  - **run():** Definisce il codice da eseguire nel thread.
  - **join():** Attende che il thread termini.
- 

## 8. Runnable

- **Descrizione:** Interfaccia funzionale che rappresenta un compito da eseguire in un thread.
  - **Principali Metodi:**
    - **run():** Metodo che contiene il codice da eseguire.
- 

## 9. Exception e RuntimeException

- **Descrizione:** Superclassi di tutte le eccezioni. **Exception** è utilizzata per gestire le eccezioni controllate, mentre **RuntimeException** è utilizzata per eccezioni non controllate.
  - **Principali Metodi:**
    - **getMessage():** Restituisce il messaggio dell'eccezione.
    - **printStackTrace():** Stampa la traccia dello stack al momento dell'eccezione.
- 

## 10. Class

- **Descrizione:** Rappresenta il runtime della classe di un oggetto. Permette di ispezionare il tipo dell'oggetto e di riflettere sulla struttura delle classi.
  - **Principali Metodi:**
    - **forName(String className):** Carica una classe per nome.
    - **getMethods():** Restituisce tutti i metodi pubblici della classe.
    - **newInstance():** Crea una nuova istanza della classe.
- 

Queste classi costituiscono la base della programmazione in Java e offrono numerosi strumenti e funzionalità per gestire aspetti fondamentali del linguaggio.

- [Object](#)
- [System](#)
- **Package**

- Class
  - ClassLoader
  - ClassValue
- 

### Classi wrapper e gestione tipi

- Boolean
  - Byte
  - Character
  - Double
  - Float
  - Integer
  - Long
  - Short
  - Void
  - Enum
- 

### Gestire le Stringhe

- [String](#)
- [StringBuilder](#)

### Gestire i numeri - Math

- [Math](#)
- StrictMath
- Number

### Altre funzionalità

- Compiler
- Process
- Runtime
- SecurityManager
- StackTraceElement

- **Thread**