

# Introduction to Programming

Rania Baghernejad

Java – part 4

## Input with Scanner

---

### مباحث

- ۲ ..... نحوه استفاده از کلاس **Scanner**
- ۲ ..... ساختار کلی استفاده از **Scanner**
- ۲ ..... دریافت ورودی از کاربر
- ۲ ..... ورودی عددی صحیح (**Integer**)
- ۲ ..... ورودی اعشاری (**Double**)
- ۳ ..... ورودی متنی (**String**)
- ۳ ..... کامل سازی ورودی ها

## کلاس Scanner و دریافت ورودی از کاربر

در برنامه‌نویسی جاوا، گاهی نیاز داریم که از کاربر ورودی بگیریم. برای این کار می‌توانیم از کلاس **Scanner** استفاده کنیم. این کلاس امکانات متنوعی برای دریافت ورودی از کاربر به صورت متنی، عددی، و دیگر انواع فراهم می‌کند.

### نحوه استفاده از کلاس Scanner

قبل از استفاده از **Scanner** باید این کلاس را وارد برنامه کنیم. برای این کار از دستور **import** استفاده می‌کنیم:

```
import java.util.Scanner;
```

پس از وارد کردن این کلاس، می‌توانیم یک شیء از آن بسازیم و از آن برای دریافت ورودی استفاده کنیم. (با مفهوم شیء و کلاس در آینده آشنا خواهیم شد.)

### ساختار کلی استفاده از Scanner

```
Scanner input = new Scanner(System.in);
```

در اینجا:

- **Scanner** نام کلاس است.
- **input** نام شیء از کلاس **Scanner**
- **System.in** نشان‌دهنده ورودی از کیبورد است.

### دریافت ورودی از کاربر

#### ورودی عددی صحیح (Integer)

برای گرفتن یک عدد صحیح از کاربر، از متد **nextInt()** استفاده می‌کنیم:

```
Scanner input = new Scanner(System.in);
System.out.println("Enter an integer: ");
int number = input.nextInt();
System.out.println("You entered: " + number);
```

#### ورودی اعشاری (Double)

برای گرفتن عدد اعشاری از کاربر، از متد **nextDouble()** استفاده می‌کنیم:

```
Scanner input = new Scanner(System.in);
System.out.println("Enter a double value: ");
```

```
double decimalValue = input.nextDouble();
System.out.println("You entered: " + decimalValue);
```

ورودی متنی (String)

برای گرفتن ورودی متنی (یک کلمه یا جمله)، از متد `nextLine()` استفاده می‌کنیم:

```
Scanner input = new Scanner(System.in);
System.out.println("Enter your name: ");
String name = input.nextLine();
System.out.println("Hello, " + name + "!");
```

**نکته:** اگر بخواهید یک کلمه را دریافت کنید (بدون اینکه نیاز به گرفتن خط کامل باشد)، از `next()` استفاده می‌کنید:

```
Scanner input = new Scanner(System.in);
System.out.println("Enter a word: ");
String word = input.next();
System.out.println("You entered: " + word);
```

نکات مهم:

توجه به `nextLine()` و `nextInt()`: زمانی که از `nextInt()` یا `nextDouble()` استفاده می‌کنید و پس از آن نیاز به گرفتن ورودی متنی دارید (مثلاً با `nextLine()`، ممکن است `nextLine()` بلافاصله یک خط خالی دریافت کند. این مشکل را می‌توان با استفاده از `input.nextLine()` پس از `nextInt()` یا `nextDouble()` حل کرد تا کاربر بتواند ورودی متنی خود را وارد کند.

```
Scanner input = new Scanner(System.in);
System.out.println("Enter an integer: ");
int number = input.nextInt();
scanner.nextLine(); // To consume the newline character
System.out.println("Enter your name: ");
String name = input.nextLine();
System.out.println("You entered number: " + number + " and name: " + name);
```

کامل‌سازی ورودی‌ها

در صورتی که نیاز به ورودی‌های متنوع از کاربر دارید، می‌توانید آن را با چندین دستور `nextInt()`، `nextDouble()`، و `nextLine()` ترکیب کنید.

مثال:

```
Scanner input = new Scanner(System.in);

System.out.print("Enter your age: ");
int age = input.nextInt();
scanner.nextLine(); // Consume newline character

System.out.print("Enter your name: ");
String name = input.nextLine();

System.out.println("Hello, " + name + ". You are " + age + " years old.");
```