The Only Interview-Ready Java Cheatsheet You'll Ever Need.

You Don't Need Any Other Cheatsheet Than This.

By @codes.learning

1/ Basic Structure:

```java
public class Main {
  public static void main(String[] args) {
    System.out.println("Hello, Java!");
  }
}

// Comments
// Single line
/* Multi-line */
```

2/ Data Types & Variables:

```java
int age = 25;
double price = 99.99;
char grade = 'A';
boolean isJavaFun = true;
String name = "Nikhil";

// Constants
final int MAX = 100;
```

By @codes.learning

5. Methods:

```java
public static int add(int x, int y) {
  return x + y;
}
```

6. Arrays:

```java
int[] nums = {1, 2, 3};
System.out.println(nums[0]);

for (int n : nums) {
  System.out.println(n);
}

// Multi-dimensional
int[][] grid = {{1,2}, {3,4}};
```

By @codes.learning

7. Object-Oriented Concepts:

```
class Car {
  String model;
  Car(String model) {
    this.model = model;
  }
  void drive() {
    System.out.println("Driving " + model);
  }
}

Car c = new Car("Tesla");
c.drive();
```

8. Inheritance & Polymorphism:

```
class Animal {
  void sound() {
    System.out.println("Some sound");
  }
}

class Dog extends Animal {
  void sound() {
    System.out.println("Bark");
  }
}
```

By @codes.learning

9. Abstraction & Interfaces:

```
abstract class Shape {
  abstract void draw();
}

class Circle extends Shape {
  void draw() {
    System.out.println("Drawing Circle");
  }
}

interface Flyable {
  void fly();
}

class Bird implements Flyable {
  public void fly() {
    System.out.println("Flying");
  }
}
```

10. Strings & Useful Methods:

```
String s = "Java";
s.length();           // 4
s.toUpperCase();      // JAVA
s.contains("va");     // true
s.charAt(1);          // a
s.equals("Java");     // true
```

By @codes.learning

11. Collections:

```java
import java.util.*;

List<String> list = new ArrayList<>();
list.add("Java");

Set<Integer> set = new HashSet<>();
Map<String, Integer> map = new HashMap<>();

for (String item : list) {
  System.out.println(item);
}
```

12. Exception Handling:

```java
try {
  int result = 10 / 0;
} catch (ArithmeticException e) {
  System.out.println("Cannot divide by zero");
} finally {
  System.out.println("Done");
}
```

By @codes.learning

13. Multithreading:

```
class MyThread extends Thread {
  public void run() {
    System.out.println("Running thread...");
  }
}

MyThread t1 = new MyThread();
t1.start();
```

14. File Handling:

```
import java.io.*;

File file = new File("data.txt");

try (BufferedReader br = new BufferedReader(new FileReader(file))) {
  String line;
  while ((line = br.readLine()) != null) {
    System.out.println(line);
  }
} catch (IOException e) {
  e.printStackTrace();
}
```

By_@codes.learning

15. Packages & Access Modifiers:

```
package myapp;

public class MyClass {} // public
private int x;                // private
protected int y;              // protected
int z;                        // default
```