

JAVA NOTES(UNIT-III,IV)

UNIT-III

Scanner class, console class (main method)

The `Scanner` class and the `Console` class are used for input/output operations in Java.

The `Scanner` class is used to read input from the user or a file. It is a class in the `java.util` package and provides various methods to read different types of input data, such as `next()`, `nextInt()`, `nextDouble()`, and more. Here is an example of using the `Scanner` class to read a user's name from the console:

```
import java.util.Scanner;
public class Main {
    public static void main(String[] args)
    {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter your name: ");
        String name = scanner.nextLine();
        System.out.println("Hello, " + name + "!");
    }
}
```

The `Console` class is used to read input and display output to the console. It is a class in the `java.io` package and provides methods to read input without echoing it to the console (such as `readPassword()`), and to display output to the console without a newline (such as `format()`). Here is an example of using the `Console` class to read a user's password from the console:

```
import java.io.Console;
public class Main {
    public static void main(String[] args) {
        Console console = System.console();
        if (console == null) {
            System.err.println("Console not available");
            System.exit(1);
        }
    }
}
```

```

    }
    pw.flush();
    // closing resources
    br.close();
    pw.close();
    System.out.println("Merged file1.txt and file2.txt into file3.txt");
}
}

```

Q4.Exception Handling and key words Related to Exception

A.Exception handling is a mechanism in Java that allows a program to handle errors and exceptional situations gracefully, rather than crashing or terminating abruptly. The Java language provides several keywords and constructs to handle exceptions, including:

1. **try-catch:** The try-catch block is used to catch and handle exceptions that are thrown by a program. It consists of a try block that contains the code that may throw an exception, and one or more catch blocks that handle the thrown exception.
2. **throw:** The throw keyword is used to explicitly throw an exception from a method or block of code.
3. **throws:** The throws keyword is used to declare that a method may throw a particular exception, and is placed in the method signature.
4. **finally:** The finally block is used to contain code that must be executed regardless of whether an exception is thrown or not. It is typically used to clean up resources such as files or database connections.
5. **catch:** The catch block is used to catch and handle exceptions that are thrown by a program. It contains code that is executed if an exception is thrown.

```

public class Example {
    public static void main(String[] args) {
        try {
            int x = 10 / 0; // this will throw an ArithmeticException
        } catch (ArithmeticException e) {
            System.out.println("Caught exception: " + e);
        } finally {
            System.out.println("Finally block executed");
        }
    }
}

```