The `java -cp` command is used to specify the classpath when running a Java application from the command line. The classpath is a list of directories and JAR (Java Archive) files that the Java Virtual Machine (JVM) uses to locate classes and resources needed by a Java program. Here's what each part of the `java -cp` command means:

- `java`: This is the command to start the Java Virtual Machine (JVM) and execute a Java program.

- `-cp` (or `-classpath`): This is a command-line option that tells the JVM to set the classpath for the Java application. You can use either `-cp` or `-classpath`; they are equivalent.

- Classpath: The classpath is a list of directories and JAR files separated by colons (on Unix-like systems) or semicolons (on Windows). It tells the JVM where to look for classes and resources when your Java program is running.

Here's an example of how you might use the `java -cp` command:

```bash

java -cp .:lib/\* com.example.MyApp

```

In this example:

- `java`: Starts the JVM.

- `-cp`: Specifies that you're setting the classpath.

- `.:lib/\*`: This is the classpath itself. It includes the current directory (`.`) and all JAR files (`\*.jar`) in the `lib` directory. The colons (`:`) separate entries on Unix-like systems. On Windows, you would use semicolons (`;`) instead of colons.

- `com.example.MyApp`: This is the name of the Java class that contains the `main` method. The JVM will execute the `main` method of this class.

By setting the classpath with `-cp`, you can ensure that the JVM can locate all the necessary classes and resources your Java program needs to run successfully.