

oops

Stack over  
LinkedList  
Tree

# JAVA

```
#include <iostream>
int main() {
    cout << "ver)
```

Source Col

```
#include <iostream>
using namespace std;
int main() {
    cout<<"Hello World";
}
```

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

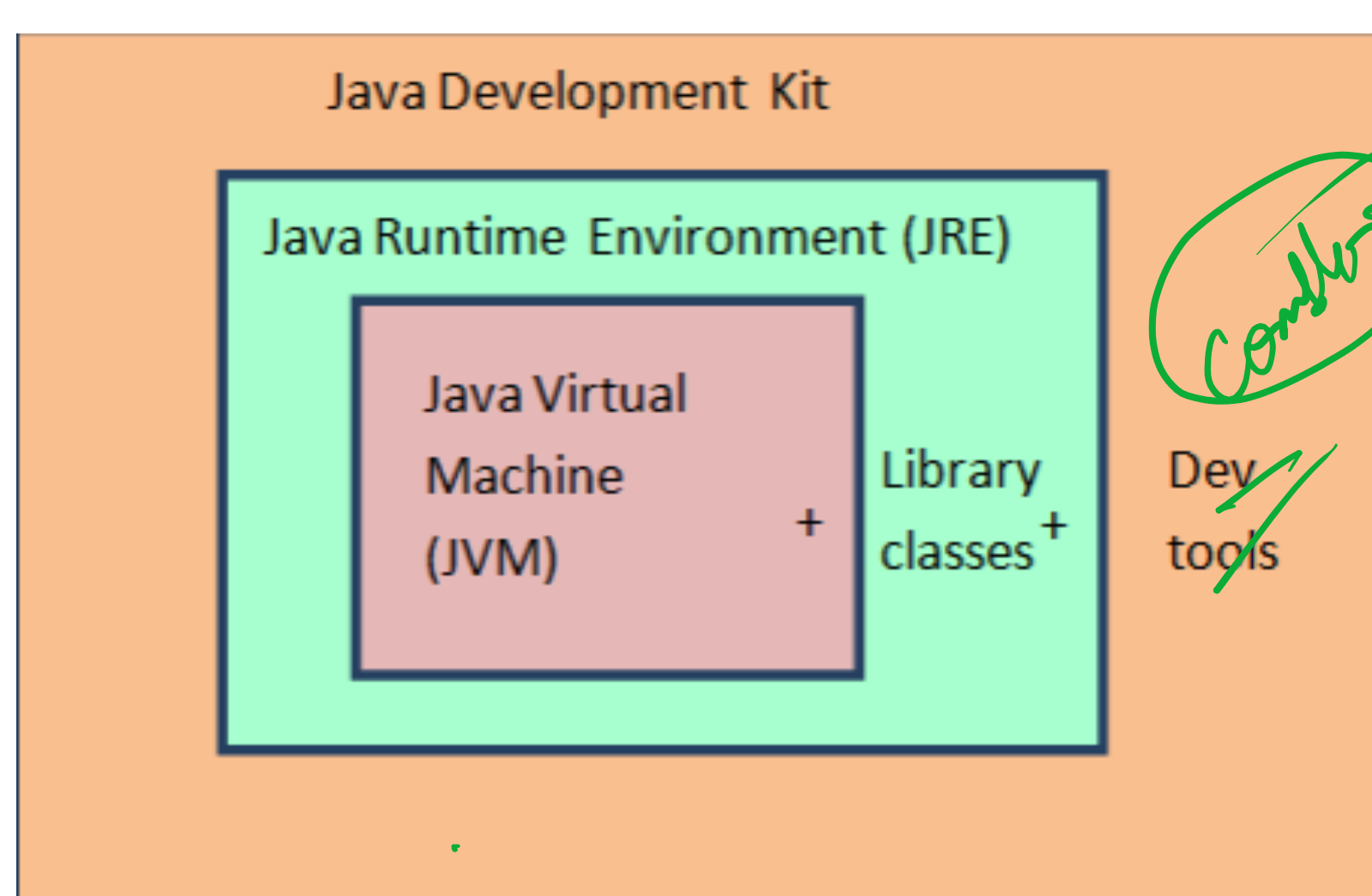

6:00

- linux
- window
- mac

2. Intervalle

6

pro winlo  
Luo



A diagram illustrating the execution flow. On the left, a green oval contains the text `*.exe`. A green arrow points from this oval to a stack of three ovals on the right. The top oval is labeled `JVM`, the middle one `Jdk`, and the bottom one `JRT`. A red arrow points from the `*.exe` oval to the `JRT` oval.

✓

A diagram illustrating the classification of words into parts of speech. A large red oval labeled "Class" has arrows pointing to four smaller blue ovals: "noun", "Adjs", "verb", and "Pun". The "Pun" oval is crossed out with a red line.

Class → Class is bulk print

5 plastic

9th Oct


$\frac{head()}{sc.next() + ()}$

new 1st CS

```
public class Student_Client {  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        Student s = new Student();  
    }  
}
```

(refine  
varich)

magn



A hand-drawn diagram of a rectangle with a horizontal line across the middle. The bottom half is labeled  $S = ar$ .

name, ask  
Introspect)

```
public static void main(String[] args) {  
    // TODO Auto-generated method stub  
    Student s = new Student();  
    System.out.println(s.name);  
    System.out.println(s.age);  
    s.Intro_yourSelf();  
    s.name = "kaju";  
    s.age = 18;  
    s.Intro_yourSelf();  
}
```

non-stati

8th C

States are

monstern

```
Person p1 = new Person("Raj", 25);
```

memory address

new two

1

person

name

age

2x

key for

SI

2