

# نحوه ی اجرای جاوا



# ابتدا جاوا را دانلود کنید

معمولا برنامه های جاوا در پنج مرحله اجرا می شوند:

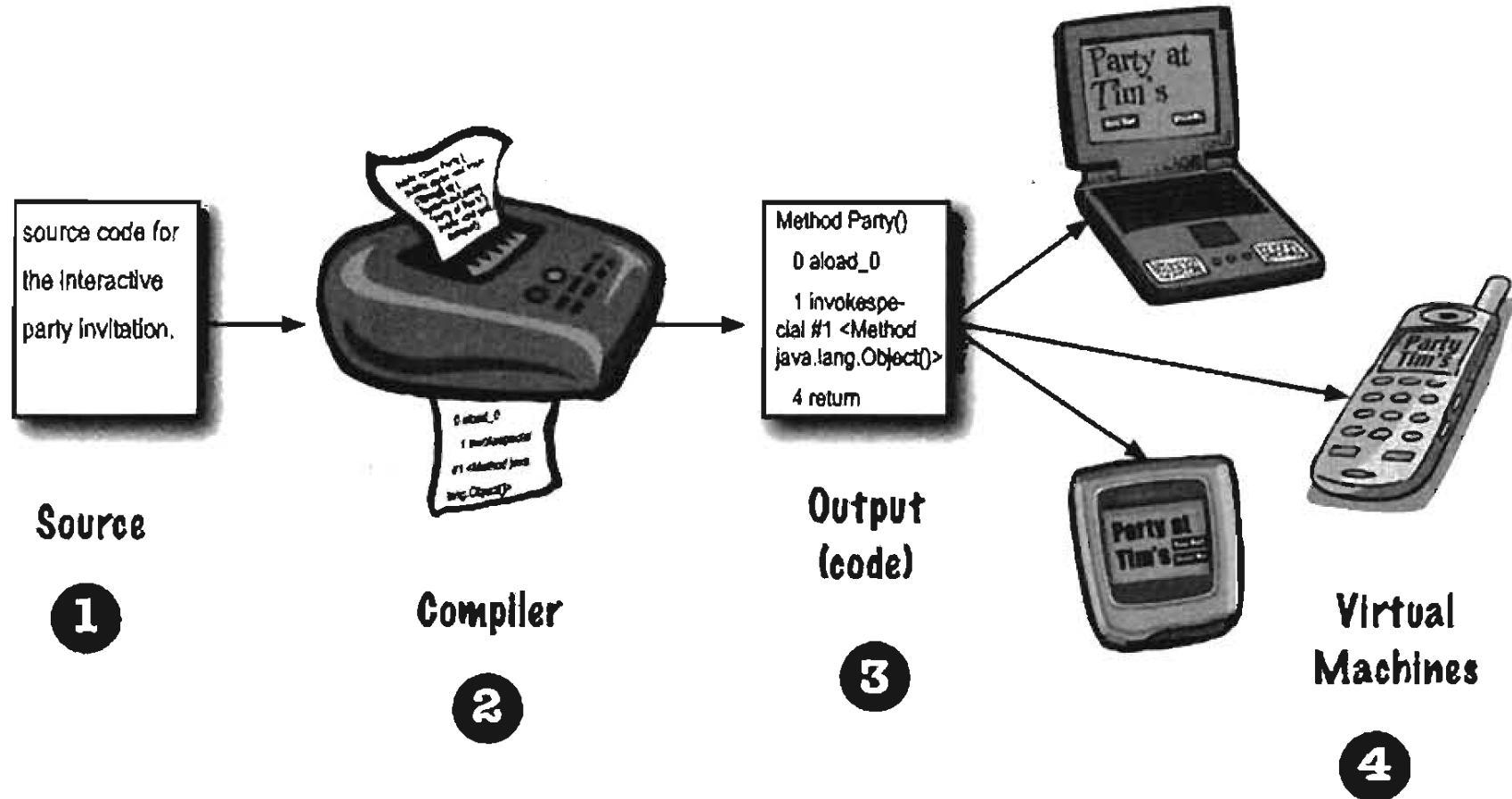
۱ - ویرایش ۲ - کامپایل ۳ - بارگذاری ۴ - بررسی ۵ - اجرا

این مراحل را از طریق موتور جاوا (Java SE Development Kit) JDK بررسی خواهیم کرد.

می توان از طریق لینک زیر جدیدترین آپدیت های JDK و مستندات آن را دانلود نمود.

[www.oracle.com/technetwork/java/](http://www.oracle.com/technetwork/java/)

# مراحل اجرای کد جاوا



# مراحل اجرای کد جاوا

```
import java.awt.*;
import java.awt.event.*;
class Party {
    public void buildInvite() {
        Frame f = new Frame();
        Label l = new Label("Party at Tim's");
        Button b = new Button("You bet");
        Button c = new Button("Shoot me");
        Panel p = new Panel();
        p.add(l);
    } // more code here...
}
```

Source

1

Party.java

```
File Edit Window Help Plead
%javac Party.java
```

Compiler

2

```
Method Party()
  0 aload_0
  1 invokespecial #1 <Method
    java.lang.Object()>
  4 return
Method void buildInvite()
  0 new #2 <Class java.awt.Frame>
  3 dup
  4 invokespecial #3 <Method
    java.awt.Frame()>
```

Output  
(code)

3

Party.class

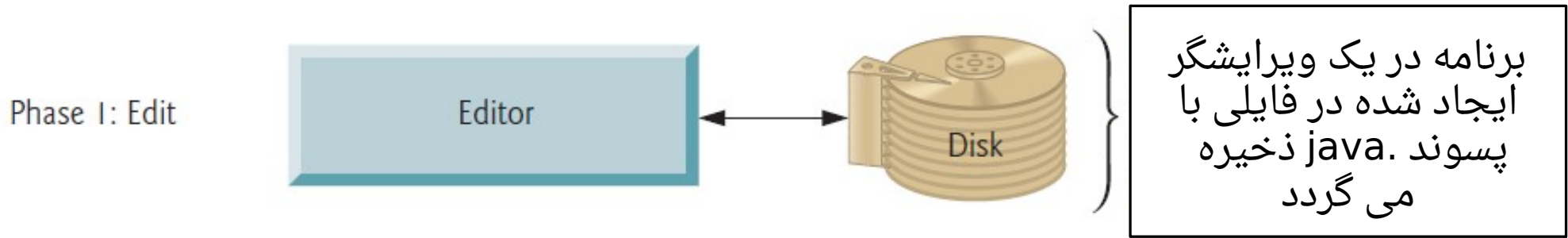


Virtual  
Machines

4





# مرحله اول: ایجاد یک برنامه

• ویرایش فایل در یک برنامه ی ویرایشگر



چند نمونه از ویرایشگر

های جاوا (IDEs)

| Eclipse  | Netbeans   | IntelliJ   | JDeveloper   |
|--|--|--|--|
|  |  |  |  |

## مرحله دوم: کامپایل برنامه جاوا به بایت کد

- از فرمان (Java compiler) `javac` برای کامپایل برنامه استفاده می شود.
- کامپایلر جاوا، کد جاوا موجود را به بایت کدهایی ترجمه می کند و بایت کدها فرمان های قابل اجرا در مرحله اجرا را مشخص می کنند.
- برخلاف بسیاری دیگر از زبان های کامپایلری که سورس کد آنها پس از کامپایل شدن به باینری یک ماشین حقیقی ترجمه می شود، برنامه های جاوا پس از کامپایل شدن به باینری ماشینی به نام ماشین مجازی جاوا یا JVM ترجمه می شود. این کد باینری را بایت کد می نامند.

## مرحله دوم: کامپایل برنامه جاوا به بایت کد

• برخلاف زبان ماشین که به سخت افزار کامپیوتر خاصی وابسته است، بایت کدها مستقل از پلت فرم هستند.

• دستورات زبان ماشین تنها بر روی همان نوع کامپیوتری که از ابتدا برای آن کامپایل شده اند، قابل اجرا می باشند اما بایت کدها میتوانند بر روی هر نوع کامپیوتری که قابلیت اجرای برنامه های جاوا را داشته باشد، اجرا شوند.

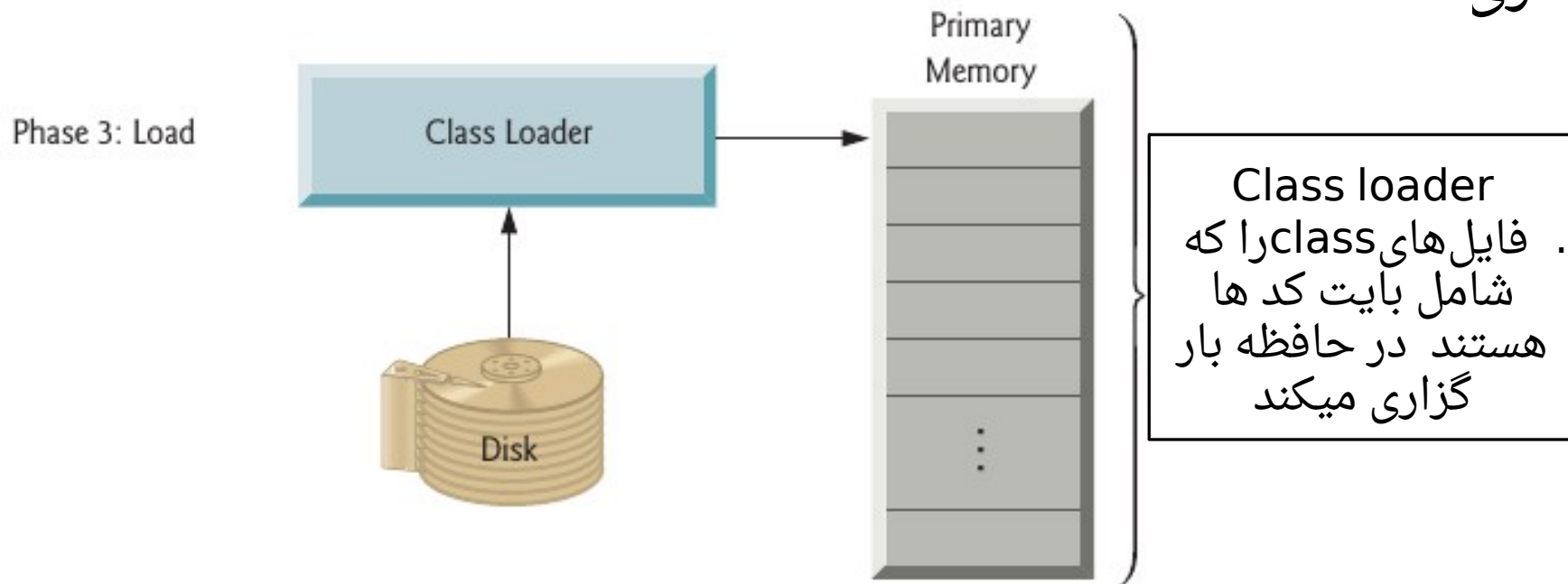
Phase 2: Compile



کامپایلر بایت کدها را  
ایجاد نموده و در فایل  
با پسوند class ذخیره  
می کند.

# مرحله سوم: بارگذاری برنامه در حافظه

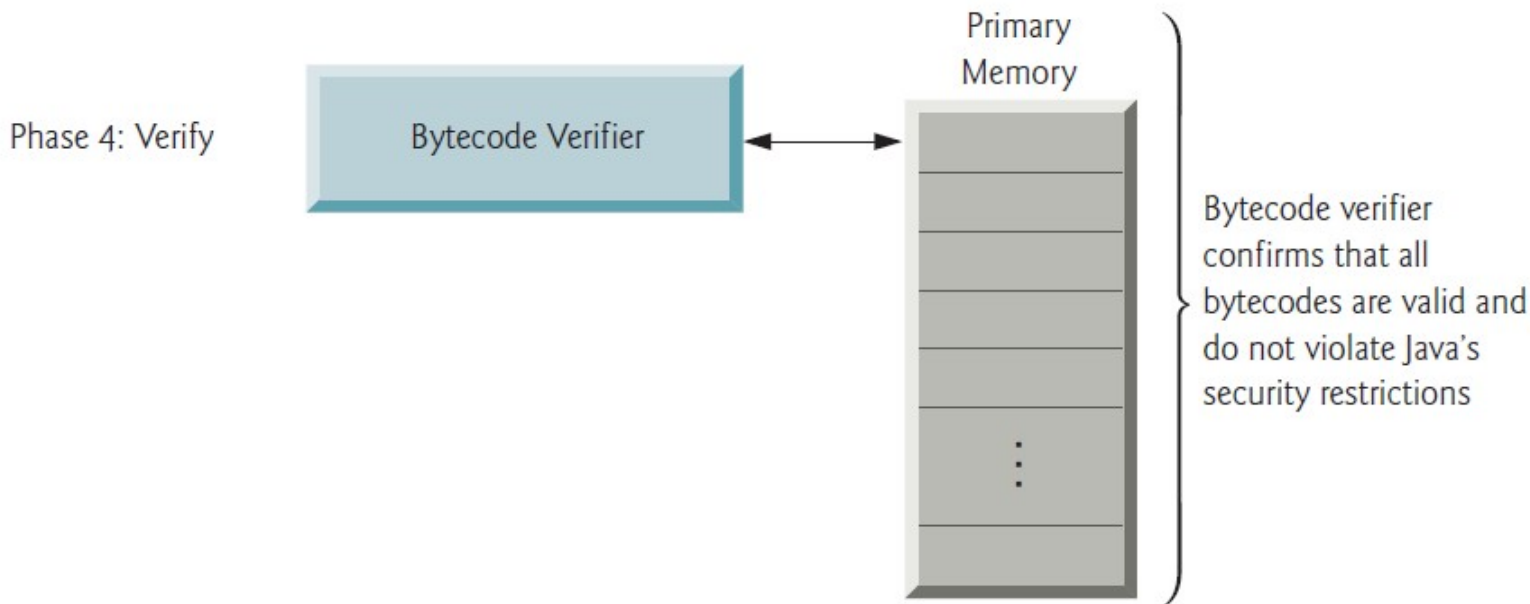
- ماشین مجازی جاوا (JVM) برنامه را جهت اجرا در حافظه قرار می دهد.  
(بارگذاری)





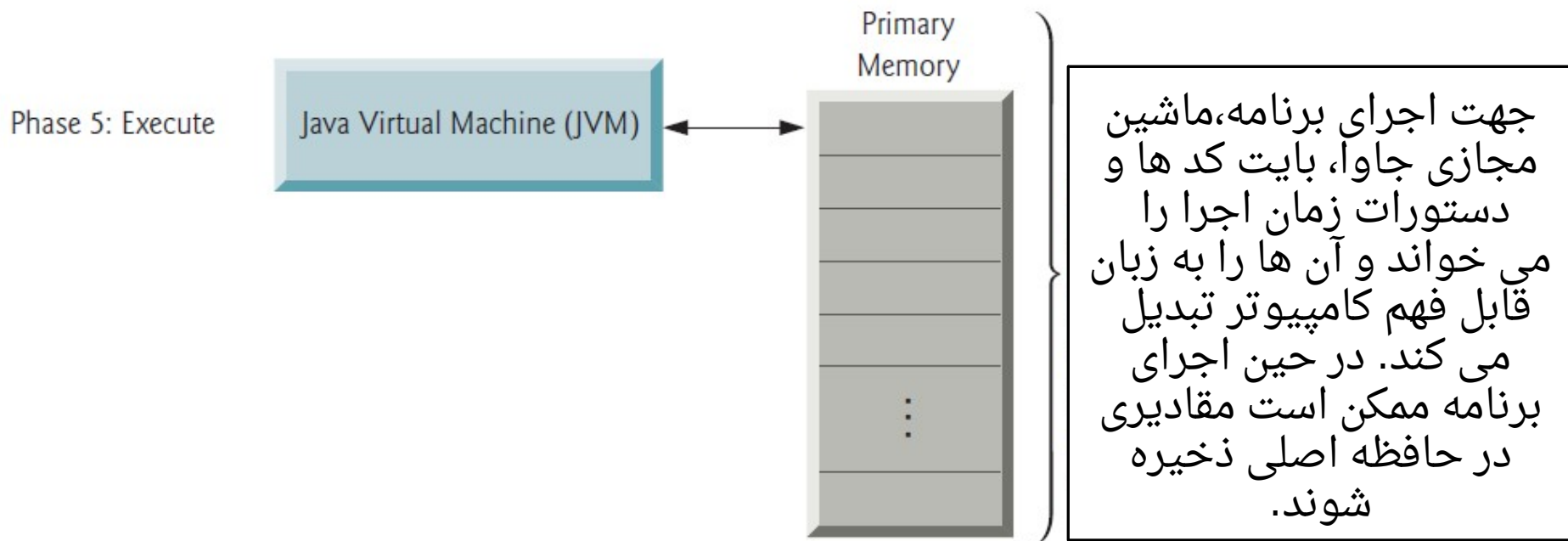
## مرحله چهارم: بررسی صحت بایت کدها

- تأیید کننده ی بایت کد بررسی می کند که بایت کدها صحیح و معتبر باشند و محدودیت های امنیتی جاوا را نقض نکنند.



# مرحله پنجم: اجرا

- ماشین مجازی جاوا برنامه را اجرا می کند و در نتیجه آن عملیات مشخص شده توسط برنامه اجرا می شود.



# مرحله پنجم: اجرا

• در نسخه های اولیه جاوا، JVM خود یک مفسر برای بایت کدهای جاوا بود و این باعث اجرای کند برنامه های جاوا می شد.

• امروزه JVM بایت کدها را از طریق ترکیبی از مفسرها و کامپایلرهای زمان اجرا (JIT)، اجرا می کنند. در این روش JVM شروع به تجزیه و تحلیل بایت کدها نموده، سپس به جست و جوی قسمت های Hot spot بایت کدها که مکررا در حال اجرا هستند می پردازد. در این قسمت ها کامپایلر زمان اجرا (JIT) که به عنوان کامپایلر HotSpot جاوا شناخته شده است بایت کدها را به زبان ماشین سطح پایین تر ترجمه می کند.

# Java Class Libraries

اکثر برنامه نویسان جاوا از مجموعه ای غنی از کلاس ها و متدهای موجود در کتابخانه ی کلاس های جاوا تحت عنوان API ها، بهره می گیرند.

API=Application Programming Interfaces

