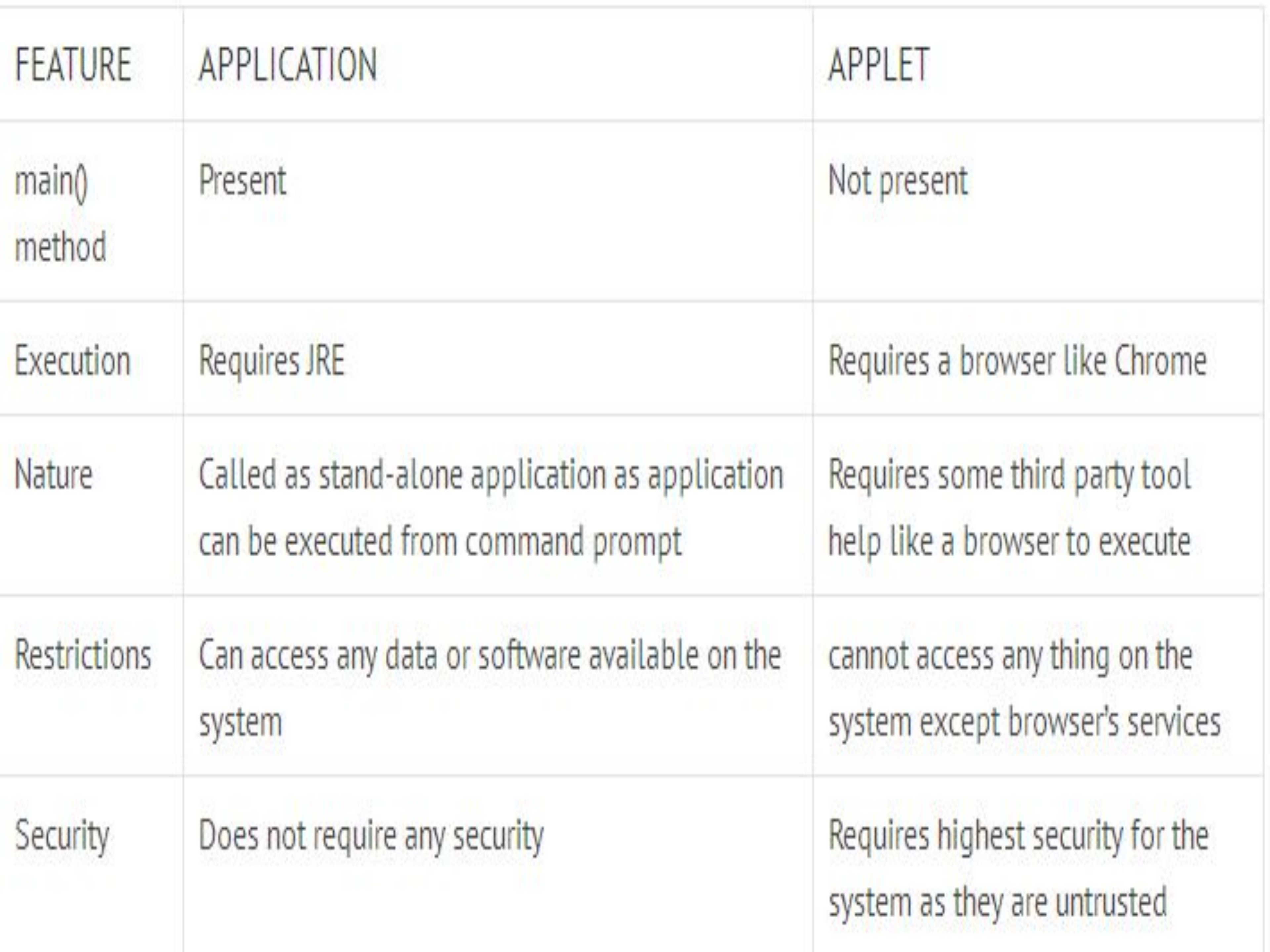
Applets

- An applet is a small Java program that is embedded and ran in some other Java interpreter program such as
  - applet viewer program called appletviewer
  - a Java technology-enabled browser
- An appletAn applet is a small Internet-based program written in Java, a programming language for the Web, which can be downloaded by any computer.
- The applet is also able to run in HTML.
- The applet is usually embedded in an HTML page on a Web site and can be executed from within a browser.



FEATURE	APPLICATION	APPLET
main() method	Present	Not present
Execution	Requires JRE	Requires a browser like Chrome
Nature	Called as stand-alone application as application can be executed from command prompt	Requires some third party tool help like a browser to execute
Restrictions	Can access any data or software available on the system	cannot access any thing on the system except browser's services
Security	Does not require any security	Requires highest security for the system as they are untrusted

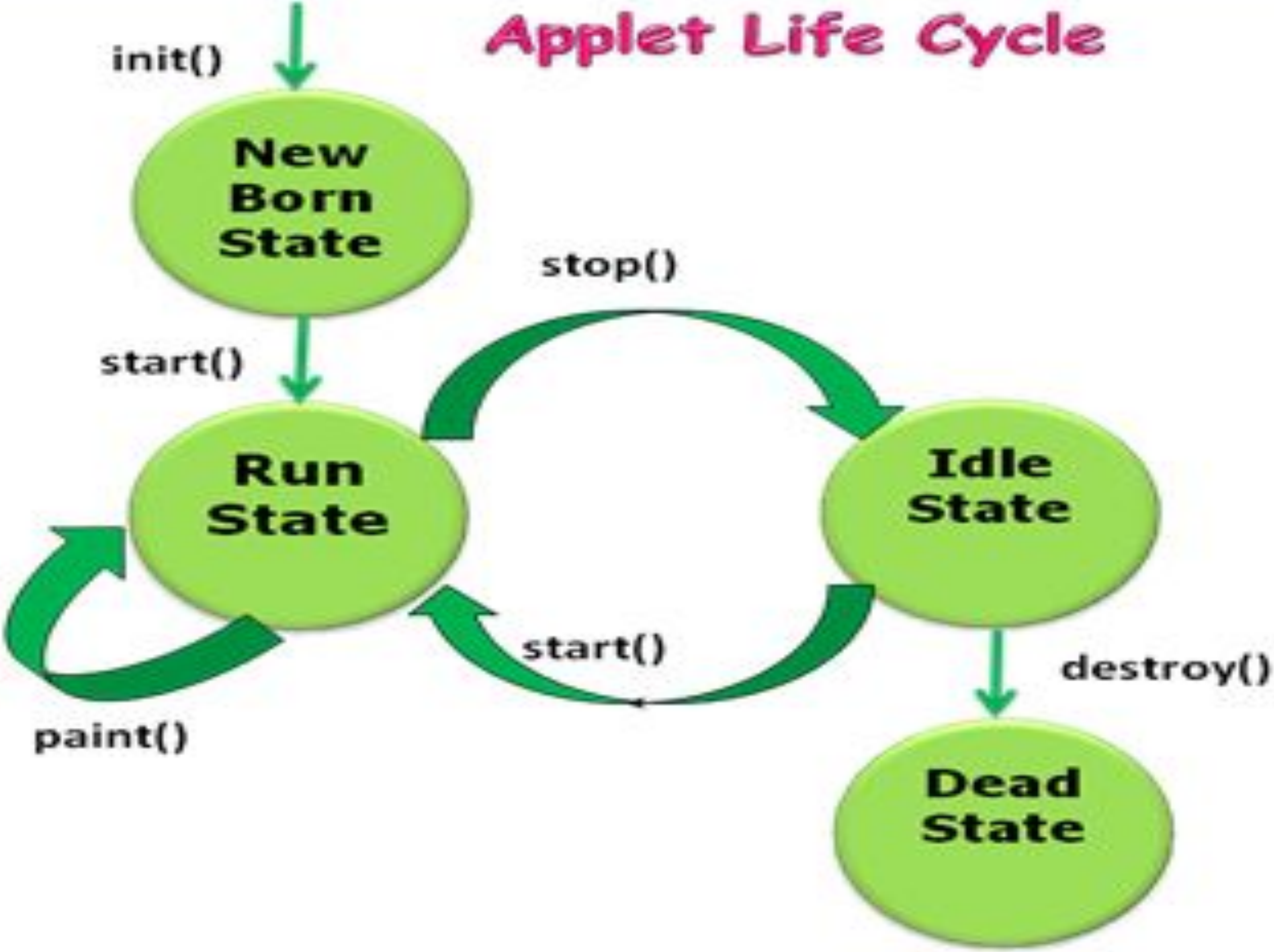
LIFE CYCLE OF APPLET

- Applet runs in the browser and its lifecycle method are called by JVM at its birth, its death and when it is momentarily away.
- Every Applet can be said to be any of the following state

  1. New Born state
  2. Running state
  3. Idle state (may or may not)
  4. Dead state

The following figure represents the life cycle of the Applet

Applet Life Cycle



init()

New Born State

start()

Run State

stop()

Idle State

start()

paint()

destroy()

Dead State

- **New Born State**
- The life cycle of an applet begins on that time when the applet is first loaded into the browser and calls the init() method.
- The init() method is called only one time in the life cycle on an applet.
- The init() method is basically called to read the PARAM tag in the html file.
- The init () method retrieves the passed parameter through the PARAM tag of html file using get Parameter() method .
- All the initialization such as initialization of variables and the objects like image, sound file are loaded in the init () method .
- After the initialization of the init() method user can interact with the Applet
- Syntax:
- **public void init()**

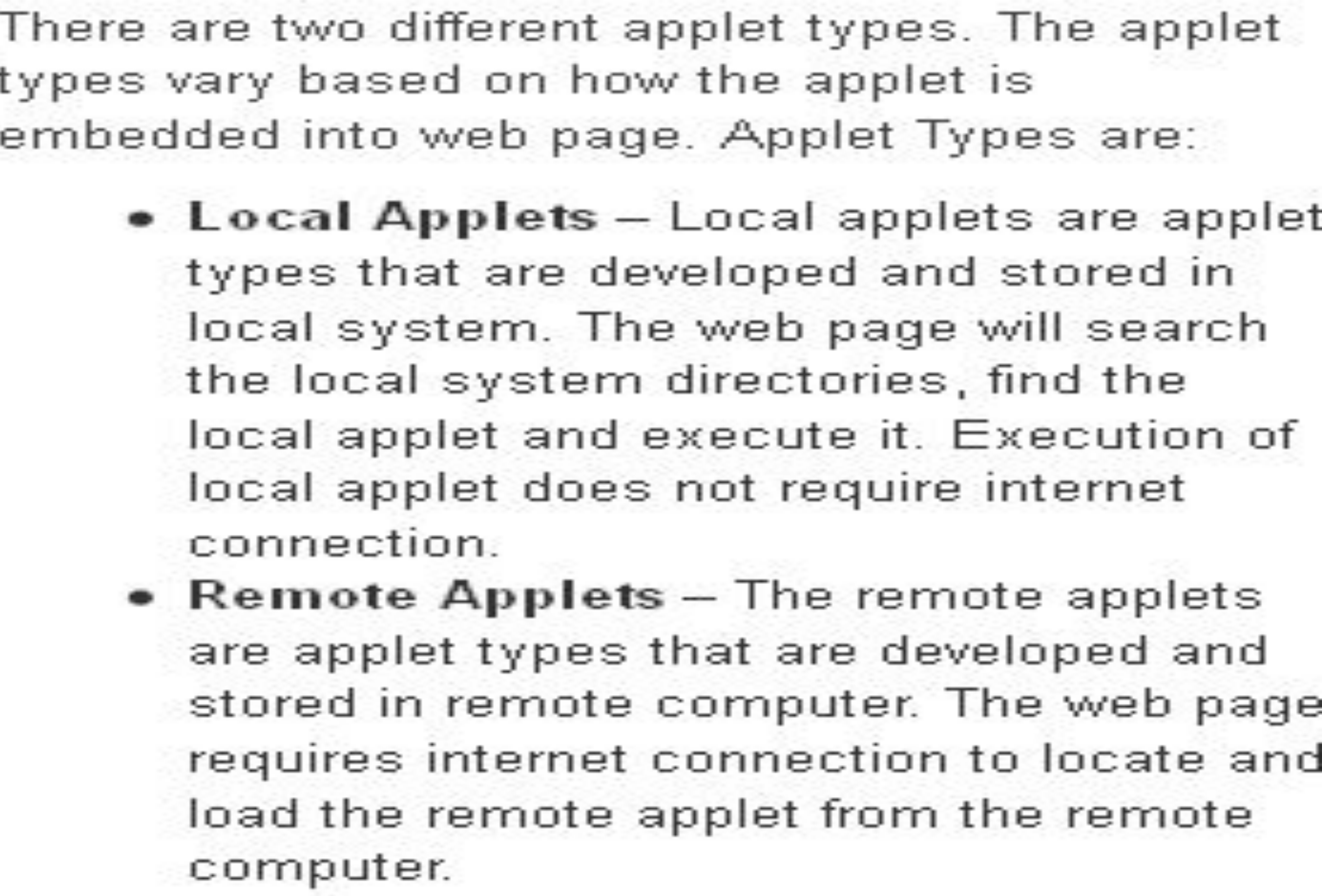
```
public void init()
{
                Statements
}
```

- **Running State**
- After initialization, this state will automatically occur by invoking the start method of applet class which again calls the run method and which calls the paint method.
- The running state also occurs from idle state when the applet is reloaded.
- This method may be called multiple times when the Applet needs to be started or restarted.
- For Example if the user wants to return to the Applet, in this situation the start() method of an Applet will be called by the web browser and the user will be back on the applet.
- In the start() method user can interact within the applet.
- Syntax:
-
  ```
  public void start()
  {
                  Statements
  }
  ```

- **Idle State**
- The idle state will make the execution of the applet to be halted temporarily.
- Applet moves to this state when the currently executed applet is minimized or when the user switches over to another page.
- At this point the stop method is invoked.
- From the idle state the applet can move to the running state.
- The stop() method can be called multiple times in the life cycle of applet
- For example the stop() method is called by the web browser on that time When the user leaves one applet to go another applet
- Syntax:
- **public void stop()**
  {
                    Statements

- **Dead State**
- When the applet programs terminate, the destroy() function is invoked which makes an applet to be in dead state.
- The destroy() method is called only one time in the life cycle of Applet like init() method.
- Syntax:
-
```
public void destroy()
{
                    Statements
}
```

- **Display State**
- The applet is said to be in display state when the paint method is called.
- This method can be used when we want to display output in the screen.
- This method can be called any number of times.
- paint() method is must in all applets when we want to draw something on the applet window.
- paint() method takes Graphics object as argument
- <u>Syntax:</u>
-
  ```
  public void paint(Graphics g)
  {
                      Statements
  }
  ```



There are two different applet types. The applet types vary based on how the applet is embedded into web page. Applet Types are:

- **Local Applets** – Local applets are applet types that are developed and stored in local system. The web page will search the local system directories, find the local applet and execute it. Execution of local applet does not require internet connection.
- **Remote Applets** – The remote applets are applet types that are developed and stored in remote computer. The web page requires internet connection to locate and load the remote applet from the remote computer.

```
import java.applet.*;
import java.awt.*;
 /* <applet code = "face" width = 300 height = 300> </applet> */
public class face extends Applet
{
 public void paint(Graphics g)
    {
      g.drawOval(100,100,100,100);
      g.fillOval(120,125,20,20);
      g.fillOval(160,125,20,20);
      g.drawLine(150,165,150,150);
      g.drawLine(130,170,170,170);
    }
}
```

- //APPLET PARAMETERS

THE HTML APPLET TAG

```
 <APPLET
[CODEBASE= codebaseURL] CODE=appletfile
[ALT=alternate text]
[NAME=applet instance name]
WIDTH=pixels HEIGHT=pixels
[ALIGN=alignment] [VSPACE=pixels][HSPACE=pixels] >
[PARAM NAME=AttributeName
   VALUE=AttributeValue]
</APPLET>
```

```
<HTML>
<HEAD>
<TITLE>Java applet example - Passing applet
parameters to Java applets</TITLE>
</HEAD>
<APPLET CODE="AppletParameterTest.class"
WIDTH="400" HEIGHT="50">
   <PARAM NAME="font"    VALUE="Dialog">
   <PARAM NAME="size"    VALUE="24">
   <PARAM NAME="string"  VALUE="Hello, world ...
it's me.   :)">
</APPLET> </HTML>
```

```
import java.applet.*;
import java.awt.*;
 public class AppletParameterTest extends Applet {
   public void paint(Graphics g) {
      String myFont   = getParameter("font");
      String myString = getParameter("string");
int mySize      =
Integer.parseInt(getParameter("size"));
      Font f = new Font(myFont, Font.BOLD, mySize);
      g.setFont(f);
      g.setColor(Color.red);
      g.drawString(myString, 20, 20);     } }
```

- **Applet – Security Limitations**
- Applets are treated as **untrusted** because they are developed by somebody and placed on some unknown Web server.
- When downloaded, they may harm the system resources or steal passwords and valuable information available on the system.
- As applets are **untrusted**, the browsers come with many security restrictions.
- Security policies are browser dependent.
- Browser does not allow the applet to access any of the system resources (applet is permitted to use browser resources, infact, applet execution goes within the browser only).

- Applets are not permitted to use any system resources like file system as they are untrusted and can inject virus into the system.
- Applets cannot read from or write to hard disk files.
- Applets should not attempt to create socket connections
- Applets cannot read system properties
- Applets cannot use any software available on the system (except browser execution area)
- Cannot create objects of applications available on the system by composition
- The JRE throws **SecurityException** if the applet violates the browser restrictions.