

# Chapter 1

## What is a computer?

A computer consists of a CPU, memory, hard disk, floppy disk, monitor, printer, and communication devices.

(( يتكون الكمبيوتر من وحدة المعالجة المركزية ، والذاكرة ، والقرص الصلب ، والقرص المرن ، الشاشة والطابعة وأجهزة الاتصال. ))

## Introduction to java:

Why using java:

Java is a general-purpose programming language.

(( Java هي لغة برمجة للأغراض العامة. ))

1. Java is an Internet programming language.

(( Java هي لغة برمجة الإنترنت. ))

2. Java can be used to develop standalone applications.

(( يمكن استخدام Java لتطوير تطبيقات مستقلة. ))

3. Java can be used to develop applications running from a browser.

(( يمكن استخدام Java لتطوير التطبيقات التي تعمل من متصفح. ))

4. Java can also be used to develop applications for hand-held devices.

(( يمكن أيضا استخدام Java لتطوير تطبيقات للأجهزة المحمولة باليد. ))

5. Java can be used to develop applications for Web servers.

(( يمكن استخدام Java لتطوير تطبيقات لخوادم الويب. ))

6. Java make with James Gosling and sun Microsystem.

## Characteristics of Java

1. Java Is Simple

Java is partially modeled on C++, but greatly simplified and improved. Some people refer to Java as "C++-" because it is like C++ but with more functionality and fewer negative aspects. ((أي أنها مشابهة لبعض الشيء لـ C++ ولكن مع بعض التعديلات على الجوانب السلبية))

2. Java Is Object-Oriented

Java is inherently object-oriented. Although many object-oriented languages began strictly as procedural languages, Java was designed from the start to be object-oriented. Object-oriented programming (OOP) is a popular programming approach that is replacing traditional procedural programming techniques.

One of the central issues in software development is how to reuse code. Object-oriented programming provides great flexibility, modularity, clarity, and reusability through encapsulation, inheritance, and polymorphism.

((جافا بطبيعتها وجوه المنحى. على الرغم من وجود العديد من اللغات الشيئية بدأت بشكل صارم كلغات إجرائية ، تم تصميم Java منذ البداية لتكون موجهة للكائنات. البرمجة الشيئية (OOP) هي طريقة برمجة شائعة تحل محل تقنيات البرمجة الإجرائية التقليدية. إحدى القضايا المركزية في تطوير البرمجيات هي كيفية إعادة استخدام الكود. توفر البرمجة الموجهة للكائنات مرونة كبيرة ونمطية ووضوحًا وقابلية لإعادة الاستخدام من خلال التغليف والوراثة وتعدد الأشكال.))

### 3. Java Is Distributed:

Distributed computing involves several computers working together on a network.

Java is designed to make distributed computing easy.

Since networking capability is inherently integrated into Java, writing network programs is like sending and receiving data to and from a file.

(( تتضمن الحوسبة الموزعة عدة أجهزة كمبيوتر تعمل معًا على شبكة. تم تصميم Java لجعل الحوسبة الموزعة سهلة. نظرًا لأن قدرة الشبكات مدمجة بطبيعتها في Java، فإن كتابة برامج الشبكة تشبه إرسال البيانات واستلامها من وإلى ملف.))

### 4. Java Is Interpreted:

You need an interpreter to run Java programs. The programs are compiled into the Java Virtual Machine code called bytecode.

The bytecode is machine-independent and can run on any machine.

that has a Java interpreter, which is part of the Java Virtual Machine (JVM).

(( أنت بحاجة إلى مترجم فوري لتشغيل برامج جافا. يتم تجميع البرامج في كود Java Virtual Machine المسمى .bytecode

الرمز الثانوي مستقل عن الآلة ويمكن تشغيله على أي جهاز

يحتوي على مترجم Java، وهو جزء من (Java Virtual Machine (JVM.))

### 5. Java Is Robust:

Java compilers can detect many problems that would first show up at execution time in other languages.

Java has eliminated certain types of error-prone programming constructs found in other languages.

Java has a runtime exception-handling feature to provide programming support for robustness.

((يمكن لمجمعي جافا اكتشاف العديد من المشكلات التي قد تظهر أولاً في وقت التنفيذ بلغات أخرى.

تخلصت Java من أنواع معينة من بنيات البرمجة المعرضة للخطأ والموجودة في لغات أخرى.

تحتوي Java على ميزة معالجة استثناءات وقت التشغيل لتوفير دعم برمجي للقوة.))

### 6. Java Is Secure:

Java implements several security mechanisms to protect your system against harm caused by stray programs.

((تنفذ Java العديد من آليات الأمان لحماية نظامك من الضرر الناجم عن البرامج الضالة.))

### 7. Java Is Architecture-Neutral:

Write once, run anywhere With a Java Virtual Machine (JVM),

You can write one program that will run on any platform.

((اكتب مرة واحدة ، وقم بتشغيله في أي مكان باستخدام (Java Virtual Machine (JVM ، يمكنك كتابة برنامج واحد يعمل

على أي نظام أساسي.))

#### 8. Java Is Portable:

Because Java is architecture neutral, Java programs are portable. They can be run on any platform without being recompiled.

((نظرًا لأن Java بنية محايدة، فإن برامج Java محمولة. يمكن تشغيلها على أي منصة دون إعادة تجميعها.))

#### 9. Java's Performance:

Java's performance Because Java is architecture neutral, Java programs are portable. They can be run on any platform without being recompiled.

((أداء Java نظرًا لأن Java هي بنية محايدة، فإن برامج Java تكون محمولة. يمكن تشغيلها على أي منصة دون إعادة تجميعها.))

#### 10. Java Is Multithreaded:

Multithread programming is smoothly integrated in Java, whereas in other languages you must call procedures specific to the operating system to enable multithreading.

((يتم دمج البرمجة متعددة الخيوط بسلاسة في Java، بينما في اللغات الأخرى، يتعين عليك استدعاء إجراءات خاصة بنظام التشغيل لتمكين تعدد مؤشرات الترابط.))

#### 11. Java Is Dynamic:

Java was designed to adapt to an evolving environment. New code can be loaded on the fly without recompilation. There is no need for developers to create, and for users to install, major new software versions. New features can be incorporated transparently as needed.

((تم تصميم Java للتكيف مع بيئة متطورة. يمكن تحميل كود جديد على الطائر دون إعادة تجميع. ليست هناك حاجة للمطورين لإنشاء إصدارات برامج رئيسية جديدة ولا يحتاج المستخدمون إلى تثبيتها. يمكن دمج الميزات الجديدة بشفافية حسب الحاجة.))

- Java Standard Edition:

J2SE can be used to develop client-side standalone applications or applets.

((هي لتطبيقات طرف المستخدم أو التطبيقات الصغيرة من طرف العميل))

- Java Enterprise Edition:

J2EE can be used to develop server-side applications such as Java servlets, Java Server Pages, and Java Server Faces.

((هي تطبيقات طرف الخادم مثل تطبيقات تصفح مواقع الانترنت أو برمجة الشبكات))

- Java Micro Edition:

J2ME can be used to develop applications for mobile devices such as cell phones.

((هي تستخدم لتطبيقات الأجهزة الصغيرة مثل تطبيقات الهاتف المحمول))

```
public static void main(String[] args)
```

```
{
```

```
System.out.println("Welcome to Java!");
```

```
}
```

((يبين هذا الامر طباعة جملة welcome to java الموجودة بين علاماتي تنصيص من اهم الامور في هذا الامر جملة

system.out.println والتي تقوم بطباعه أي جملة او متغير موجود داخلها بشرط ان تكون الجملة بين علاماتي

تنصيص وكما يظهر لنا راس العملية method والذي يتكون من modifier , datatype , methodname , and

attribute والذي يرمزون لعدة رموز سيتم ذكرهم لاحقا

#### Anatomy of a Java Program:

1. Class name
2. Main method
3. Statements
4. Statement terminator
5. Reserved words

## 6. Comments

## 7. Blocks

((هنا يتم شرح كل ما يخص تقسيمات الأوامر في جافا لنفترض المثال السابق حيث يوجد method main وكما يسمى الاقتران الرئيسي أي انه هو الأساس او البداية في عملية تنفيذ البرنامج))  
((أيضا يوجد statement مثل جملة الطباعة System.out.print وهي من أهم الأوامر في java))  
((Statement Terminator و يقصد بها نهاية الجملة أي الفاصلة المنقوطة))  
((Reserved words و اما هذه فيقصد بها الأمور التعريفية مثل data type، modifier، etc.))  
((block وهذه المقصود بها بداية العملية و نهايتها و توجد في أماكن عدة مثل الشروط و العمليات و التكرار و غيرها من الأمور الموجودة في لغة البرمجة Java))

### Programming Style and Documentation:

1. Appropriate Comments (وضع التعليقات)
2. Naming Conventions (التسميات المناسبة)
3. Proper Indentation and Spacing Lines  
((تباعداً الأماكن بين الأوامر و يقصد بها جعل مسافة سطرين على الأقل بين كل امر واخر))
4. Block Styles  
((وهنا يقصد وضع الاقواس الخاصة في بداية الأوامر وبدايتها في سطور منفصلة وتقسيمها بحيث تكون واضحة لمن يريد رؤية البرنامج))

### Programming Errors:

- 1- Syntax error
- 2- Runtime error
- 3- Logic error

# Chapter 2

## Reading Input from the Console:

```
Scanner input = new Scanner(System.in);
```

((هذا الامر هو امر الادخال في لغة البرمجة Java بحيث انك تقوم بانشاء object with tybe scanner و تقوم أيضا بوضع متغير اسمه system.in داخل الاقواس المقررة بال argument بالطبع يجب ان تقوم بعمل جملة طباعة لتخبر المستخدم او الشخص الاخر عن ما تريد منه إدخاله كما في الامر القادم))

```
System.out.print("Enter a double value: ");
```

```
Scanner input = new Scanner(System.in);
```

```
double d = input.nextDouble();()
```

An identifier is a sequence of characters that consist of

letters, digits, underscores (\_), and dollar signs (\$).

An identifier must start with a letter, an underscore (\_),

or a dollar sign (\$). It cannot start with a digit .

An identifier cannot be a reserved word. (See Appendix

A, "Java Keywords," for a list of reserved words.)

An identifier cannot be true, false, or

null.

An identifier can be of any length.

((هذه النقاط تشرح قواعد تسمية المتغيرات من أهمها عدم التسمية بالكلمات المحجوزة او كلمات false، null و لا باس بموضوع الطول الخاص بالمعرف))

**Variables:**

**Int radius = 1;**

↓      ↓  
**Data      variables**

**Type      name**

Conversion Rules:

1. If one of the operands is double, the other is converted into double.  
((إذا كان أحد المعاملين مزدوجًا، فسيتم تحويل الآخر إلى مضاعف.))
2. Otherwise, if one of the operands is float, the other is converted into float.  
((خلاف ذلك، إذا كان أحد المعاملين عائمةً، فسيتم تحويل الآخر إلى تعويم.))
3. Otherwise, if one of the operands is long, the other is converted into long.  
((خلافًا لذلك، إذا كان أحد المعاملين طويلًا، فسيتم تحويل الآخر إلى طويل.))
4. Otherwise, both operands are converted into int.  
((خلاف ذلك، يتم تحويل كلا المعاملين إلى int.))

**Casting:**

int i = (int)3.0;

((يقصد بهذا المصطلح تحويل المتغير من نوع يأخذ حيز كبير في منطقة صغيرة و في هذا المثال يتم وضع المتغير من نوع double الذي مساحته 64 بت في مكان من نوع int الذي مساحته 32 بت ويتم ذلك عن طريق كتابة نوع المتغير المراد بين الاقواس))

## Chapter 9

OOP(object oriented program):

involves programming using objects. An object represents an entity in the real world that can be distinctly identified. For example, a student, a desk, a circle, a button, and even a loan can all be viewed as objects. An object has a unique identity, state, and behaviors. The state of an object consists of a set of data fields (also known as properties) with their current values. The behavior of an object is defined by a set of methods.

((و يقصد بها انه يتم بها برمجة الأشياء الحقيقية على هيئة أجزاء هذه الأجزاء يتم تعريفها بعمليات object الذي يتم فيها وضع الخصائص الأساسية لكل شيء))

### Classes:

Classes are constructs that define objects of the same type. A Java class uses variables to define data fields and methods to define behaviors. Additionally, a class provides a special type of methods, known as constructors, which are invoked to construct objects from the class.

((هي تترجم بمعنى الفئات أي التقسيمات التي تحدد الكائنات من نفس النوع ليتم استدعائها فيما بعد))

### Constructors:

Constructors are a special kind of methods that are invoked to construct objects.

((هناك بعض الشروط لهذا النوع من الأوامر مثل كتابته على مرحلتين و هذا ما يعرف ب overloading و الغرض منه تعريف المتغيرات الموجودة في هذا الكائن class ))

### The Random Class:

You have used Math.random() to obtain a random double value between 0.0 and 1.0 (excluding 1.0). A more useful random number generator is provided in the java.util.Random class.

((في هذا الامر هو احد الأوامر الموجودة في Java و الذي يجعل البرنامج يخرج مخرجات عشوائية ويوجد فيها method لكل نوع من البيانات مثل nextInt(), nextLong(long n) وغيرها من method ))

## Chapter 10

### Class Abstraction and Encapsulation:

Class abstraction means to separate class implementation from the use of the class. The creator of the class provides a description of the class and let the user know how the class can be used. The user of the class does not need to know how the class is implemented. The detail of implementation is encapsulated and hidden from the user.

((و ملخص كل هذا الكلام انه يتم تجزئة الكائن وجعله افضل و امن وهذا ما يسمى بعملية encapsulation و احد الأشياء التي تقوم بها في هذه العملية جعل المتغيرات privet و جعل ال method ك public ))

### Aggregation or Composition:

#### 1- Aggregation Between Same Class:

Aggregation may exist between objects of the same class.

((أي انه يتم عمل هذه العلاقة بعمل متغير من نفس نوع class داخل نفس class ))

## Chapter 11

### Superclasses and Subclasses:

The special form of software consists of a main object and a sub-object where the child object consists of all the properties of the main object in addition to its own properties.

((يتكون الشكل الخاص للبرمجيات من كائن رئيسي وكائن فرعي حيث يكون الكائن الفرعي من جميع الخصائص للكائن الرئيسي بالإضافة الى خصائصه الخاصة.))

### Are superclass's Constructor Inherited?

No. They are not inherited.

They are invoked explicitly or implicitly.

Explicitly using the super keyword.

A constructor is used to construct an instance of a class.

Unlike properties and methods, a superclass's constructors are not inherited in the subclass. They can only be invoked from the subclasses' constructors, using the keyword super. If the keyword super is not explicitly used, the superclass's no-arg constructor is automatically invoked.

(( لا يتم توريث constructor الخاص بالكائن الرئيسي و يعبر عنه بكلمه super و لا يتم استدعاء constructor ))

### Superclass's Constructor Is Always Invoked

A constructor may invoke an overloaded constructor or its superclass's constructor. If none of them is invoked explicitly, the compiler puts super() as the first statement in the constructor.

((قد يستدعي المُنشئ مُنشئ مُحمل بشكل زائد أو مُنشئ الطبقة الفائقة الخاص به. إذا لم يتم استدعاء أي منها صراحةً ، فإن المترجم يضع super () كأول بيان في المُنشئ.))

### Overriding Methods in the Superclass:

A subclass inherits methods from a superclass. Sometimes it is necessary for the subclass to modify the implementation of a method defined in the superclass. This is referred to as method overriding.

((بمعنى انه في بعض الأحيان يمكن ان يعدل بعض الأمور في نفس object و لهذا يتم استعمال طريقة Overriding Methods ))

### Polymorphism:

Polymorphism means that a variable of a supertype can refer to a subtype object.

((يعني تعدد الأشكال أن متغيراً من supertype يمكن أن يشير إلى كائن من النوع الفرعي))

### Visibility Modifiers:

- 1- **Public** : use ant where in your program.
- 2- **Privet** : use just in same class.
- 3- **Protected**: use in same class and same package.
- 4- **Default**: not use in sub class but you can use in same package.

## Chapter 12

### The ArrayList Class:

You can create an array to store objects. But the array's size is fixed once the array is created. Java provides the ArrayList class that can be used to store an unlimited number of objects.

((يمكنك إنشاء مصفوفة لتخزين الأشياء. لكن حجم المصفوفة ثابت بمجرد إنشاء المصفوفة. توفر Java فئة ArrayList التي يمكن استخدامها لتخزين عدد غير محدود من الكائنات.))

### Generic Type:

ArrayList is known as a generic class with a generic type E. You can specify a concrete type to replace E when creating an ArrayList. For example, the following statement creates an ArrayList and assigns its reference to variable cities. This ArrayList object can be used to store strings.

((أي انه بإمكانك تخزينها على هيئة سلاسل و تخزين المعلومات على هذا الأساس وسيتم تفصيلها لاحقاً))

## Chapter 13

### Exception Advantages:

Now you see the advantages of using exception handling.

It enables a method to throw an exception to its caller.

Without this capability, a method must handle the exception or terminate the program.

((الآن ترى مزايا استخدام معالجة الاستثناءات. إنه يمكن أسلوبًا من طرح استثناء لمن يتصل به. بدون هذه الإمكانية ، يجب على الطريقة معالجة الاستثناء أو إنهاء البرنامج.))

### Exception Types:

#### 1- Exception:

- a- **ClassNotFoundException.**
- b- **IOException.**
- c- **RuntimeException.**
  - i- **ArithmeticException**
  - ii- **NullPointerException**
  - iii- **IndexOutOfBoundsException**
  - iv- **IllegalArgumentException**

#### 2- Error:

- a- **LinkageError.**
- b- **VirtualMachineError.**

All this type in Throwable interface.

## Chapter 14

What is Generics?

Generics is the capability to parameterize types.

With this capability, you can define a class or a method with generic types that can be substituted using concrete types by the compiler. For example, you may define a generic stack class that stores the elements of a generic type. From this generic class, you may create a stack object for holding strings and



a stack object for holding numbers. Here, strings and numbers are concrete types that replace the generic type.

((يقصد انه هذا الطريقة تؤدي الي تجميع التكديس او التجميعات للبيانات ويمكن استبدالها من نفس المترجم ))

Why generic?

The key benefit of generics is to enable errors to be detected at compile time rather than at runtime. A generic class or method permits you to specify allowable types of objects that the class or method may work with. If you attempt to use the class or method with an incompatible object, a compile error occurs.

((تسمح لك الفئة أو الطريقة العامة بتحديد الأنواع المسموح بها من الكائنات التي قد تعمل الفئة أو الطريقة معها. إذا حاولت استخدام الفئة أو الأسلوب مع كائن غير متوافق ، فسيحدث خطأ في الترجمة.))

### Erasure and Restrictions on Generics

Generics are implemented using an approach called type erasure. The compiler uses the generic type information to compile the code, but erases it afterwards. So the generic information is not available at run time. This approach enables the generic code to be backward-compatible with the legacy code that uses raw types.

((لا تتوفر المعلومات العامة في وقت التشغيل. يمكن هذا الأسلوب الكود العام من أن يكون متوافقاً مع الإصدارات السابقة مع الكود القديم الذي يستخدم الأنواع الأولية.))

## Chapter 15

**The Thread Class:**

**Start :**

**isAlive:**

**The isAlive() method is used to find out the state of a thread. It returns true if a thread is in the Ready, Blocked, or Running state; it returns false if a thread is new and has not started or if it is finished**

((وهذا يعني انه يتأكد ان العملية قيد التنفيذ او بشكل اخر على قيد الحياه ام لا))

**setPriority:**

**join:**

You can use the join() method to force one thread to wait for another thread to finish

((تستخدم هذه الطريقة لانجاز المهام بشكل كامل قبل الدخول للمهمة الأخرى))

**sleep:**

The sleep(long mills) method puts the thread to sleep for the specified time in milliseconds.

((هذه الطريقة تستخدم لوضع المهمة المعينة بوضع السكون لفترة بالمللي ثانية))

**yield:**

**You can use the yield() method to temporarily release time for other threads.**

((تستخدم لتحرير الوقت من المؤشرات الأخرى أي التأخير مدة بسيطة لإنجاز مهمة أخرى قبل الدخول في المهمة التي بعدها))

**interrupt:**

**The interrupt() method interrupts a thread in the following way: If a thread is currently in the Ready or Running state, its interrupted flag is set; if a thread is currently blocked, it is awakened and enters the Ready state, and an java.io.InterruptedException is thrown.**

((تقوم طريقة interrupt () بمقاطعة مؤشر ترابط بالطريقة التالية: إذا كان مؤشر ترابط في حالة الاستعداد أو التشغيل حاليًا ، يتم تعيين علامة المقاطعة الخاصة به ؛ إذا تم حظر مؤشر ترابط حاليًا ، فسيتم تنبيهه ويدخل في حالة الاستعداد ، ويتم طرح ((.java.io.InterruptedException

**Deadlock**

Sometimes two or more threads need to acquire the locks on several shared objects.

This could cause deadlock, in which each thread has the lock on one of the objects and is waiting for the lock on the other object.

((بمعنى انه في بعض الأحيان قد يصل المترحم الى طريق مسدود في احد المهام فيقوم بارسال رسالة للمستخدم يوضح له ذلك))

**The synchronized keyword**

To avoid race conditions, more than one thread must be prevented from simultaneously entering certain part of the program, known as critical region.

((لتجنب ظروف السباق ، يجب منع أكثر من مؤشر ترابط واحد من الدخول في وقت واحد إلى جزء معين من البرنامج ، والمعروف باسم المنطقة الحرجة.))