**Thư viện toán học Math trong Java**

**1. Tổng quát**

Thư viện toán học Math là thư viện cung cấp các phương thức có sẵn để giải quyết các vấn đề về số học. Các phương thức này phần lớn trả về giá trị kiểu double. Bảng dưới đây sẽ liệt kê tên và công dụng của các phương thức có trong thư viện này.

| **Tên phương thức** | **Công dụng** |
| --- | --- |
| PI | Trả về giá trị PI. |
| abs(a) | Trả về giá trị tuyệt đối của số a. |
| max(a,b) | Trả về giá trị lớn nhất giữa 2 số a và b. |
| min(a,b) | Trả về giá trị nhỏ nhất giữa 2 số a và b. |
| sqrt(a) | Trả về căn bậc 2 của số a. |
| pow(x,y) | Tính lũy thừa xy. |
| sin(radian) | Tính sin, với radian = Math.PI \* góc / 180. |
| cos(radian) | Tính cos. |
| tan(radian) | Tính tan. |

Để sử dụng các phương thức trên, chúng ta sẽ thực hiện theo cú pháp sau:

**Cú pháp**

|  |  |
| --- | --- |
| 1 | Math.[Tên\_phương\_thức]; |

Sau đây chúng ta sẽ tìm hiểu về các loại phương thức này.

**2. Các phương thức của Math**

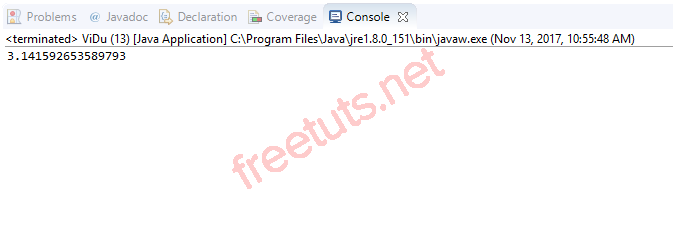
### **2.1. Math.PI**

Math.PI là phương thức trả về giá trị số PI.

**Ví dụ**

|  |  |
| --- | --- |
| 1  2  3 | public static void main(String[] args) {      System.out.println(Math.PI);    // 3.141592653589793  } |

Kết quả sau khi biên dịch chương trình:



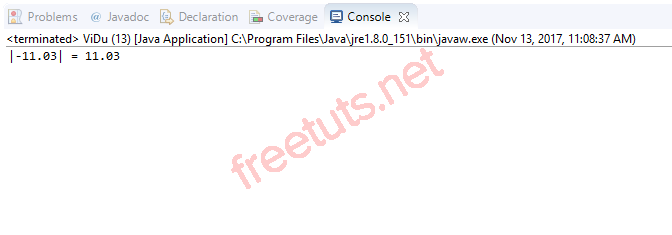
### **2.2. Math.abs(a)**

Math.abs(a) là phương thức trả về giá trị tuyệt đối của số a. Số a này có thể có kiểu dữ liệu là int, double, float hoặc long.

**Ví dụ**

|  |  |
| --- | --- |
| 1  2  3  4 | public static void main(String[] args) {      float floatNumber = Math.abs(-11.03f);      System.out.println("|-11.03| = " + floatNumber);  } |

Kết quả sau khi biên dịch chương trình:



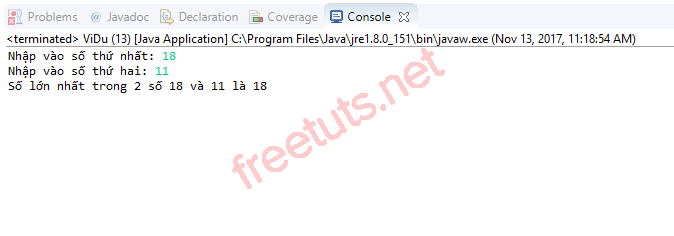
### **2.3. Math.max(a,b)**

Math.max(a,b) là phương thức trả về giá trị lớn nhất giữa 2 số a và b.

**Ví dụ**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | public static void main(String[] args) {      int firstNumber, secondNumber;      Scanner scanner = new Scanner(System.in);        System.out.print("Nhập vào số thứ nhất: ");      firstNumber = scanner.nextInt();      System.out.print("Nhập vào số thứ hai: ");      secondNumber = scanner.nextInt();        // tìm số lớn nhất giữa 2 số firstNumber và secondNumber      // sử dụng phương thức Math.max()      int maxNumber = Math.max(firstNumber, secondNumber);      System.out.println("Số lớn nhất trong 2 số " + firstNumber + " và " + secondNumber +          " là " + maxNumber);  } |

Kết quả sau khi biên dịch chương trình:



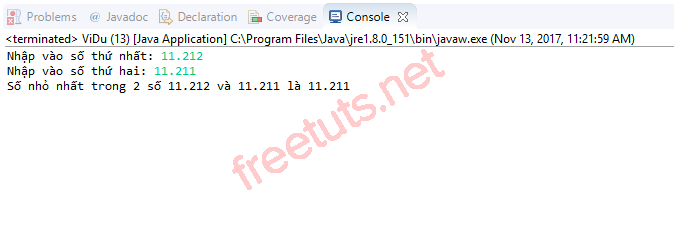
### **2.4. Math.min(a,b)**

Math.min(a,b) là phương thức trả về giá trị nhỏ nhất giữa 2 số a và b.

**Ví dụ**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | public static void main(String[] args) {      double firstNumber, secondNumber;      Scanner scanner = new Scanner(System.in);        System.out.print("Nhập vào số thứ nhất: ");      firstNumber = scanner.nextDouble();      System.out.print("Nhập vào số thứ hai: ");      secondNumber = scanner.nextDouble();        // tìm số nhỏ nhất giữa 2 số firstNumber và secondNumber      // sử dụng phương thức Math.min()      double minNumber = Math.min(firstNumber, secondNumber);      System.out.println("Số nhỏ nhất trong 2 số " + firstNumber + " và " + secondNumber +          " là " + minNumber);  } |

Kết quả sau khi biên dịch chương trình:



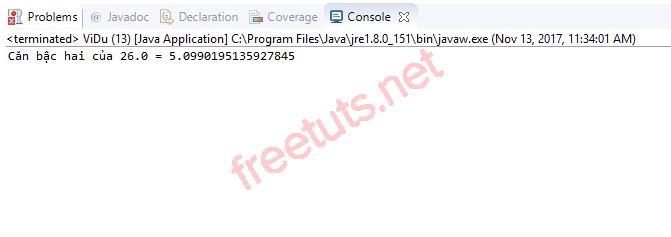
### **2.5. Math.sqrt(a)**

Math.sqrt(double a) là phương thức trả về căn bậc 2 của số a. Kết quả của phương thức này sẽ trả về số căn bậc hai của số a có kiểu double.

**Ví dụ**

|  |  |
| --- | --- |
| 1  2  3  4  5 | public static void main(String[] args) {      double number = 26;      double canBacHai = Math.sqrt(number);      System.out.println("Căn bậc hai của " + number + " = " + canBacHai);  } |

Kết quả sau khi biên dịch chương trình:



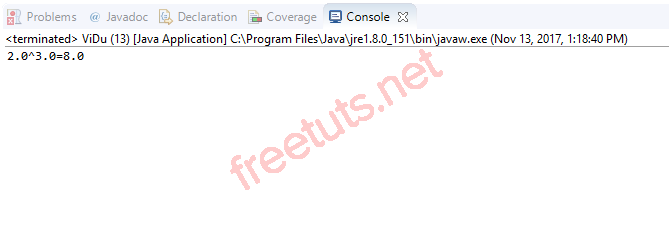
### **2.6. Math.pow(x,y)**

Math.pow(x,y) là phương thức dùng để tính tính lũy thừa xy(x là cơ số và y là số mũ).

**Ví dụ**

|  |  |
| --- | --- |
| 1  2  3  4  5 | public static void main(String[] args) {      double x = 2, y = 3, z;      z = Math.pow(x, y);      System.out.println(x + "^" + y + " = " + z);  } |

Kết quả sau khi biên dịch chương trình:



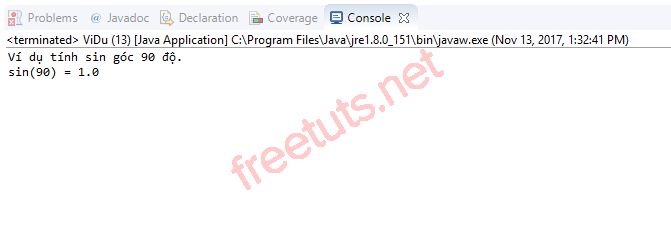
### **2.7. Math.sin(radian)**

Math.sin(double radian) là phương thức dùng để tính sin, với số **radian = Math.PI \* góc / 180**.

**Ví dụ**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | public static void main(String[] args) {      System.out.println("Ví dụ tính sin góc 90 độ.");      int goc = 90;        // đưa 1 góc về radian      double radian = Math.PI \* goc / 180;      System.out.println("sin(" + goc +") = " + Math.sin(radian));  } |

Kết quả sau khi biên dịch chương trình:



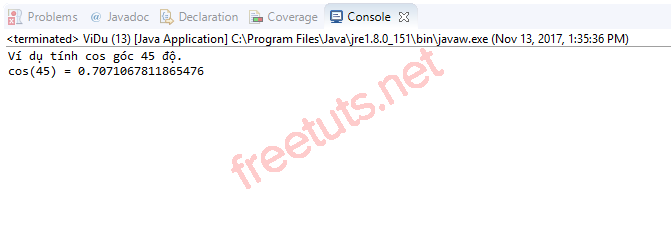
### **2.8. Math.cos(radian)**

Math.cos(double radian) là phương thức dùng để tính cos, với số **radian = Math.PI \* góc / 180**.

**Ví dụ**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | public static void main(String[] args) {      System.out.println("Ví dụ tính cos góc 45 độ.");      int goc = 45;        // đưa 1 góc về radian      double radian = Math.PI \* goc / 180;      System.out.println("cos(" + goc +") = " + Math.cos(radian));  } |

Kết quả sau khi biên dịch chương trình:



### **2.9. Math.tan(radian)**

Math.tan(double radian) là phương thức dùng để tính tan, với số **radian = Math.PI \* góc / 180**.

**Ví dụ**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | public static void main(String[] args) {      System.out.println("Ví dụ tính tan góc 45 độ.");      int goc = 45;        // đưa 1 góc về radian      double radian = Math.PI \* goc / 180;      System.out.println("tan(" + goc +") = " + Math.tan(radian));        // ngoài ra chúng ta có thể tính tan = sin/cos      double tan = Math.sin(radian) / Math.cos(radian);      System.out.println("tan(" + goc +") = " + tan);  } |

Kết quả sau khi biên dịch chương trình:

