Que. Explain Array in java.

An array is a group of like-typed variables that are referred to by a common name. Arrays of any type can be created and may have one or more dimensions. A specific element in an array is accessed by its index. Arrays offer a convenient means of grouping related information.

One-Dimensional Arrays :

➔Syntax : type var-name[ ];

➔Example: int month_days[]; //**month_days is an array variable, no array actually exists**.

➔The **value of month_days is set to null**, which represents an array with no value.

➔we **must have to use** ‘**new’** for physical array

month_days = new int[12]; // **new** is a special operator that **allocates memory.**

➔We **must specify the type and number of elements** to allocate.

➔**Array allocated by new will automatically be initialized to zero.**

➔Obtaining an array is a two-step process.

- First, you must **declare a variable** of the desired array type.
  int month_days[];
- Second, you must **allocate the memory**
  month_days = new int[12];

➔**In Java all arrays are dynamically allocated.** All array **indexes start at zero.**

➔It is possible to combine the declaration of the array variable with the allocation of the array itself:

For example: int month_days[] = new int[12];

➔Arrays can be initialized when they are declared.

For example: int month_days[] = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };

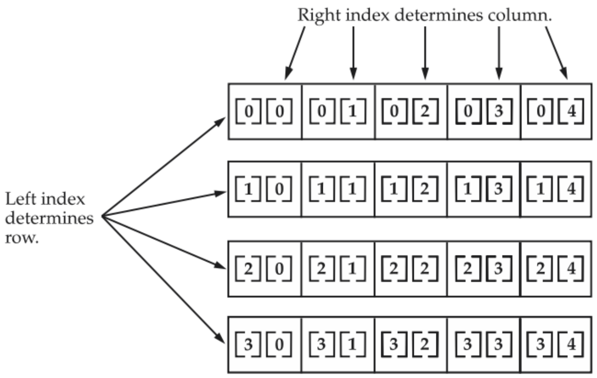
Here, the array will automatically be created large enough to hold the number of elements you specify in the array initializer. There is **no need to use new.**

Multidimensional Arrays :

In Java, multidimensional arrays are actually **arrays of arrays. (Asst. Prof. Viral Patel)**

- **Two-dimensional array :**

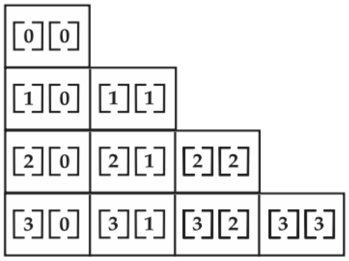
```
int twoD[][] = new int[4][5];
        Or
int twoD[][] = new int[4][];
twoD[0] = new int[5];
twoD[1] = new int[5];
twoD[2] = new int[5];
twoD[3] = new int[5];
```



In multidimensional arrays the length of each array is under our control.

For example, In two dimensional array in which the sizes of the second dimension are unequal.

```
int twoD[][] = new int[4][];
twoD[0] = new int[1];
twoD[1] = new int[2];
twoD[2] = new int[3];
twoD[3] = new int[4];
```



- **Three-dimensional array :**
  Example :   int threeD[][][] = new int[3][4][5];

Alternative Array Declaration Syntax :

➔The following declarations are equivalent:

```
int al[] = new int[3];
int[] a2 = new int[3];
```

➔The following declarations are equivalent:

```
char twod1[][] = new char[3][4];
char[][] twod2 = new char[3][4];
```

Array of Object :

For example, We can create three objects of class EMP by using array of object concept.

```
EMP e[] = new EMP[3];  //create only three references e[0], e[1] and e[2].
e[0]=new EMP();   // create first object and give reference e[0]
e[1]= new EMP();  // create second object and give reference e[1]
e[2]= new EMP();  // create third object and give reference e[2]
```

OR

```
EMP e[] = new EMP[3];  // create three references e[0], e[1] and e[2].
for(int i=0;i<3;i++)
{
  e[i] = new EMP(); // directly create three objects using 'for loop'
}
```