PASCAL - ARRAY OF POINTERS

Pascal allows defining an array of pointers. There may be a situation, when we want to maintain an array, which can store pointers to integers or characters or any other data type available. Following is the declaration of an array of pointers to an integer −

```
type
   iptr = ^integer;
var
   parray: array [1..MAX] of iptr;
```

This declares *parray* as an array of MAX integer pointers. Thus, each element in *parray*, now holds a pointer to an integer value. Following example makes use of three integers, which will be stored in an array of pointers as follows −

```
program exPointers;
const MAX = 3;
type
   iptr = ^integer;

var
   arr: array [1..MAX] of integer = (10, 100, 200);
   i: integer;
   parray: array[1..MAX] of iptr;

begin
   (* let us assign the addresses to parray *)
   for i:= 1 to MAX do
   parray[i] := @arr[i];

   (* let us print the values using the pointer array *)
   for i:=1 to MAX do
      writeln(' Value of arr[', i, '] = ' , parray[i]^ );
end.
```

You can also use an array of pointers to string variables to store a list of strings as follows −

```
program exPointers;
const
   MAX = 4;
type
   sptr = ^ string;

var
   i: integer;
   names: array [1..4] of string = ('Zara Ali', 'Hina Ali', 'Nuha Ali','Sara Ali') ;
   parray: array[1..MAX] of sptr;

begin
   for i := 1 to MAX do
      parray[i] := @names[i];

   for i:= 1 to MAX do
      writeln('Value of names[', i, '] = ' , parray[i]^ );
end.
```

When the above code is compiled and executed, it produces the following result −

```
Value of names[1] = Zara Ali
Value of names[2] = Hina Ali
Value of names[3] = Nuha Ali
Value of names[4] = Sara Ali
```