

# Dart – Reverse a List

## Dart – Reverse a List

---

You can reverse a List in Dart either by using built-in functions or using looping statements with swap mechanism.

Reversing a List makes the first element to be placed at last position, second element placed at last but one position and so on.

A special case would be, when you reverse a list of numbers sorted in ascending order, the resulting reversed list would be in descending order.

In this tutorial, we will some of the methods to reverse a Dart List.

### Reverse Dart List using List.reversed

---

List.reversed returns an Iterable of the objects in this list in reverse order. We can use this iterable to initialize a new List.

In the following Dart program, we take a list and initialize a new list with the constructor `new List.from()` and pass the reversed list iterable as argument to the constructor.

#### Dart Program

```
void main(){
  //a list
  var myList = [24, 56, 84, 92];

  //intialize a new list from iterable to the items of reversed order
  var reversedList = new List.from(myList.reversed);

  print(reversedList);
}
```

#### Output

```
D:\tutorialkart\workspace\dart_tutorial>dart example.dart
[92, 84, 56, 24]
```

The elements of the list are reversed.

## Reverse Dart List in-place by Swapping in For Loop

---

We can use a for loop to iterate till the middle of the list and for each iteration we swap the element at the index with the element at the `N-1-index`.

As we doing this in the original list, the original list will finally have the reversed list. Hence, this is called reversing a Dart List in-place.

In the following Dart Program, we shall reverse a list in-place using for loop and swapping.

### Dart Program

```
void main(){
  var myList = [24, 56, 84, 92];

  for(var i=0;i<myList.length/2;i++){
    var temp = myList[i];
    myList[i] = myList[myList.length-1-i];
    myList[myList.length-1-i] = temp;
  }

  print(myList);
}
```

### Output

```
D:\tutorialkart\workspace\dart_tutorial>dart example.dart
[92, 84, 56, 24]
```

The original list, after execution of for loop, contains the reversed list.

## Reverse Dart List with Reversed List as a new List and Saving Original

---

Also, we can reverse a list in some primitive way. Where we create an empty list of size as that of original list, and copy the elements one by one from the original list from the end to the new list from starting.

### Dart Program

```
void main(){
  var myList = [24, 56, 84, 92];

  var reversedList = new List(myList.length);

  for(var i=0;i<myList.length;i++){
    reversedList[i] = myList[myList.length-1-i];
  }

  print(reversedList);
}
```

```
        reversedList[i] = myList[myList.length-1-i];
    }
    print(reversedList);
}
```

### Output

```
D:\tutorialkart\workspace\dart_tutorial>dart example.dart
[92, 84, 56, 24]
```

## Conclusion

In this [Dart Tutorial](#), we learned how to reverse a list using List.reversed, for loop with swapping, and a primitive way where we copy elements from original to reversed.

### Dart Programming

- ◆ [Dart Tutorial](#)
- ◆ [Install Dart on Windows](#)
- ◆ [Dart - Hello World](#)
- ◆ [Dart - Variables](#)
- ◆ [Dart - Comments](#)
- ◆ [Dart - If Else](#)
- ◆ [Dart - For Loop](#)

### Dart String Operations

- ◆ [Dart - Concatenate Strings](#)
- ◆ [Dart - Split String](#)
- ◆ [Dart - Replace Substring in String](#)
- ◆ [Dart - Find Substring of String](#)
- ◆ [Dart - String Length](#)
- ◆ [Dart - Trim String](#)

### Dart Exception Handling

- ◆ [Dart - Try Catch](#)

## Dart List Operations

◆ Dart - List

◆ Dart List - Iterate

◆ Dart - Check if List is Empty

◆ Dart - Check if List Contains Element

⇒ **Dart Reverse List**

◆ Dart Join Lists

◆ Dart - Check Equality of Two Lists