Async: we use this method to pre-execute a lengthy task in background.

import 'dart:io';  
  
void main() {  
 performtasks();  
}  
  
void performtasks() {  
 task1();  
 task2();  
 task3();  
}  
  
void task1() {  
 String result = 'task1 data';  
 print('task 1 completed');  
}  
  
void task2() {  
 Duration d = Duration(seconds: 10);  
 Future.delayed(d, () {  
 String result = 'task2 data';  
 print('task 2 completed');  
 });  
}  
  
void task3() {  
 String result = 'task3 data';  
 print('task 3 completed');  
}

now using await command in a situation where we have to wait for the return value of a function.

import 'dart:io';  
  
void main() {  
 performtasks();  
}  
  
void performtasks() async {  
 task1();  
 String r = await task2();  
 task3(r);  
}  
  
void task1() {  
 String result = 'task1 data';  
 print('task 1 completed');  
}  
  
Future<String> task2() async {  
 Duration d = Duration(seconds: 5);  
 String result = 'null';  
 await Future.delayed(d, () {  
 result = 'task2 data';  
 print('task 2 completed');  
 });  
 return result;  
}  
  
void task3(String resultof2) {  
 String result = 'task3 data';  
 print('task 3 completed \n $resultof2');  
}