Difference between method overriding and overloading in Dart.

Ans:

|  |  |  |
| --- | --- | --- |
| Aspect | Method overriding | Method overloding |
| Defiition | Method overriding occurs when a subclass provides a specific implementation for a method already defined in its superclass. The overridden method in the subclass has the same name, return type, and parameters as the superclass method. | Method overloading occurs when multiple methods in the same class have the same name but different parameters (different numbers or types of parameters). |
| Inheritance  Requirement | Overriding requires a superclass-subclass relationship, where the subclass inherits the method from its superclass. | Overloading can occur in a single class, and it doesn't necessarily involve inheritance. |
| Return Type | The return type of the overriding method in the subclass must be the same as or a subtype of the return type in the superclass method. | The return type can be the same or different for overloaded methods. |
| Method  Signature | The overriding method must have the same name and parameter types as the method being overridden in the superclass. | Overloaded methods have the same name but may have different parameter types, numbers, or both. |
| Use of  ‘@override’ | It is recommended to use the annotation before the method in the subclass that is intended to override the superclass method. | The annotation is not used for overloaded methods. |
| Superclass  Call | In the subclass, the ‘super’ keyword can be used to call the superclass's version of the method. | Overloaded methods do not have a superclass - subclass relationship, so there is no concept of a superclass call. |
| Example | class Superclass {  void method();  }  class Subclass extends Superclass{  @override  void method() {  supermethod() ;  print( 'Subclass method');  }} | class Calculator {  int add(int a,int b){  return a+b;  }  Double add (double a,double b){  return a + b;  }  } |