

# Overview of Copy Constructor Solutions

- Explain what a copy constructor is
  - It is another form of the constructor which is used to create a new object and initialize it from an existing object of the same class
  - It is also used to pass arguments and return values into and out of a function call by value
- What is the prototype of the copy constructor?
  - `T(const T& other);`            `// Copy constructor for type T`

- How is it invoked?
  - It is automatically invoked when we create an object with an initial value
    - `T a(...);`           // Calls constructor (not the copy constructor!)
    - `T b = a;`           // Traditional syntax
    - `T c{a};`           // Alternative syntax from C++11
  - The compiler adds code which calls the copy constructor with the appropriate argument
  - The copy constructor is also invoked when passing to and returning from a function by value

- Explain why it is not normally necessary to implement a copy constructor when writing a class
  - If we do not provide a copy constructor, the compiler will generate a default copy constructor which
  - Copies data members which are built-in types
  - Calls the copy constructor of members which are classes
- In what circumstances is it necessary?
  - When the default is not good enough
  - Usually this is when the class manages a resource