

## Activity 2

1. **Static array** are arrays that have fixed size.

Eg. `Int a[5]`; this has size of 5.

- They are allocated memory in compile time.
- No need to delete static arrays, they are deleted automatically after going out of scope.
- Allocated memory from the stack.

**Dynamic Arrays**

- Dynamic Arrays are allocated on heap.
- Size of dynamic arrays can be determined either at compilation or at run-time (flexible).
- You can construct very large dynamic arrays on heap, unlike static arrays.
- You need to manually delete dynamic arrays after you no longer need them.

## Advantage of dynamic array

- Flexibility of size
  - Saving of memory space
  - preventing of memory wastage found in static arrays
2. When a pointer is pointing at the memory address of a variable but after some time that variable is deleted from that memory location while the pointer is still pointing to it, then such a pointer is known as a dangling pointer.
  3. Pointer is mainly used for dynamic data allocation or data in the fly.
  4. Array and pointers are closely related to each other. It can be used to find out or bring out the elements and more used to in multidimensional arrays. But it is mostly up to the programmer to decide where to use it.
  5. If you use pointer arrays, you would have the flexibility to add more elements in your pointer array and also you can decrease the number of elements in the pointer array to save much more free space for other things in your program. Because of that, pointer arrays also called as dynamic arrays. And other reason is to stop using indexes and manipulate the information needed.
  6. Constraints of smart pointers are:
    - Performance limitation slower than the traditional pointers
    - Memory leaks still will exist
    - smart pointers don't help against loops in graph-like structures.
  7. Which parameter passing is more advantageous and why?
    - There are three possible parameters passing they are pass by value, pass by pointer and pass by reference.
    - It usually depends based on the programmer and what kind of programs they are developing but passing by value is more preferred because the value usually stays the same but different address can give different value.

8. `shared_ptr` to `unique_ptr` is not allowed. once you have turned lifetime management of resource over to `shared_ptr` there is no changing of mind.

9. Discuss the main features introduced in C++ versions (C++11– C++20).

- Lambda Expressions: A lambda expression lets you define functions locally, at the place of the call, thereby eliminating much of the tedium and security risks that function objects incur.
- three-way comparison using the "spaceship operator", operator `<=>`
- immediate functions using the new `constexpr` keyword
- Allowing attributes for namespaces and enumerators
- UTF-8 (u8) character literals
- Initializers in if and switch statements
- `std::any`, for holding single values of any type
- In C++03, there are restrictions on what types of objects can be members of a union. For example, unions cannot contain any objects that define a non-trivial constructor or destructor. C++11 lifts some of these restrictions
- And many other features.