

Lecture 4

Thursday, February 13, 2020 12:13 AM

- Use class to work on the currency conversion problem
 - homework 2
- CurrencyFactory
 - eager initialization
 - create all currency objects in default constructor
 - then use curr_type to return the object
- Automatic objects / lazy initialization
 - destroy objects when no longer needed--> free store objects
 - using the keyword **new**
 - create obj, return address of the obj
 - use * or -> to access free store obj
 - Currency* c = **new** Currency("USD", 1.0);
 - (*c).GetSymbol();
 - c->SetExchangeRate(0.95);
 - delete c;
 - be aware of memory leak.

```
class CurrencyFactory
{
public:
    CurrencyFactory();
    Currency* GetCurrency(int currencyType);
private:
    Currency* currencies_[5];
};
```
 - efficiency : no longer
 - default construct Currency objects
 - I assign Currency objects
 - I copy construct Currency objects
 - delete free store objects

```
CurrencyFactory::~~CurrencyFactory()
{
    for (int i=0; i<5; ++i)
    {
        delete currencies_[i];
    }
}
```
- templates
 - allow us to write functions and classes with types as parameters.