## - From the Chapter 11 Video Notes:

- In the 5th video note (Section 11.4.4), the narrator provides so additional information on the use of the protected access modifier. Can methods use protected as an access modifier?

***No, methods are either public or private. Data can be protected.***

They can be defined as protected

Program comments:

**Account class**

- Credit and Debit were to have been defined as methods and not as properties.

- Looking ahead to the change that is needed for how the Debit method is to be defined in the CheckingAccount class, you should have:

public virtual void Credit(decimal value)

{

// set

// {

if (value > 0)

Balance += value;

else

throw new ArgumentOutOfRangeException("Balance", value,

"Balance can't be negative, 0 or greater please.");

// }

}

// public Decimal Debit

public virtual bool Debit(decimal value)

{

// set

// {

if (value < Balance)

{

Balance -= value;

return true;

}

else

{

throw new ArgumentOutOfRangeException("Limit Exceeded", value,

"Debit amount exceeded account balance.");

return false;

}

// }

}

**SavingsAccount**

Ok

**CheckingAccount**

The Credit method was to have invoked the base class’s version of this method and then included the fee, as follows:

public override void Credit(decimal value)

{

base.Credit(value);

Balance -= feeAmount;

// set

// {

// value = value - feeAmount;

// if (value > 0)

// Balance += value;

// else

// throw new ArgumentOutOfRangeException("Balance", value,

// "Balance can't be negative, 0 or greater please.");

// }

}

The Debit method was too to have invoked the base class’s method:

// public virtual Decimal Debit

public override bool Debit(decimal value)

{

if (base.Debit(value))

// set

{

Balance -= feeAmount;

return true;

//value = value + feeAmount;

// if (value < Balance)

// {

// Balance -= value;

}

else {

// throw new ArgumentOutOfRangeException("Limit Exceeded", value,

// "Debit amount exceeded account balance.");

return false;

}

}

}

**In the Test Program**

- You are very generous to give a savings account a 50% interest rate (5% is even high .05)

- Because of changes to the Credit and Debit methods, there will need to be changes, such as:

// ac2.Credit = ac2.CalculateInterest();

ac2.Credit(ac2.CalculateInterest());

- You should also create a CheckingAccount object and use several of the methods