# Assignment (due: 2016/10/26)

Black-Scholes 가격 계산기 클래스를 구현하고, 주어진 옵션의 가격을 계산하는 프로그램을 작성하시오.

**(1) Option 클래스**

- 추상클래스로 PlainVanillaOption과 DigitalOption의 base class임
- 멤버함수
  - ✓ double npv() = 0
  - ✓ void setPricingMethod(PricingMethod) : 가상함수 아님
  - ✓ void setMarketVariables(MarketVariable) : 가상함수 아님
- 멤버변수
  - ✓ 옵션의 잔여만기(double)
  - ✓ 옵션의 행사가격(double)
  - ✓ call / put을 구분하는 옵션타입
  - ✓ pricingMethod,
  - ✓ marketVariables
- npv는 가격(double)을 리턴하는 함수이며, setPricingMethod와 setMarketVarialbles는 void 함수로 각각 pricingMethod와 marketVariables를 변경 또는 입력하는 함수임
- pricingMethod는 Analytic, MonteCarlo, BinomialTree 의 세 유형의 enum 변수이며, 생성자에서 pricingMethod를 Analytic으로 초기화함
- 생성자는 옵션의 잔여만기(double)와 옵션의 행사가격(double), call / put을 구분하는 옵션타입을 입력 받음

**(2) PlainVanillaOption 클래스**

- Option 클래스에서 상속받은 derived class임
- Option의 순수가상함수를 Plain-vanilla 옵션에 대해 구현함

**(3) DigitalOption 클래스**

- Option 클래스에서 상속받은 derived class임
- Option의 순수가상함수를 Digital 옵션에 대해 구현함

**(4) MarketVariables 클래스**

- 주가, 금리, 배당, 변동성의 시장변수를 멤버변수로 가지는 클래스
- set 함수와 get 함수를 구현함
- Option 클래스에서 MarketVariable을 hsa-a 관계로 접근함

**(5) void calcPrice(Option&, PricingMethod)** 함수에서 인자로 입력된 Plain-Vanilla 옵션 또는 Digital 옵션의 가격을 계산하고 출력하도록 구현함

(Option& 의 위치에 PlainVanillaOption 또는 DigitalOption의 객체를 입력할 수 있으며, over-riding 가상함수를 통해 해당 옵션의 가격 함수를 호출할 수 있음)

**(6) main함수**

아래의 옵션을 종류에 따라 instance를 생성하고 calcPrice 함수를 통해 가격을 출력하시오.

| 옵션 종류 | 타입 | 행사가 | 만기 |
|---|---|---|---|
| PV | Call | 250 | 0.1 |
| PV | Put | 260 | 0.2 |
| DG | Put | 240 | 0.15 |
| PV | Call | 270 | 0.25 |
| DG | Call | 230 | 0.2 |

시장데이터: 주가=250, 금리=0.02, 배당=0.01, 변동성=0.2

☆ 가급적 중복된 코드를 사용하지 않도록 함.