JAVA Programming

Project #3 –Hierarchy , Interface, Abstract

Hanju Kim Dankook University

2019 Spring 05- 02

Java Programming: Hierarchy, Interface, Abstract / free Hanju Kim 32141318

1 Project Introduction:

이번 과제는 좀더 객체지향 프로그래밍 적인 프로그래밍을 작성하는 것으로서 객체지향 프로그래밍이란 프로그래밍에서 필요한 데이터를 추상화 시켜 상태와 행위를 가진 객체를 만들고 그 객체들 간의 유기적인 상호작용을 통해 로직을 구성하는 프로그래밍을 말한다. 따라서 가장 큰 장점으로 대형 프로젝트를 작성할때 업무를 분담하기가 쉽다는게 장점입니다.

객체지향 프로그래밍은 5가지 키워드를 가지고 있습니다.

1. 클래스+객체
2. 추상화
3. 캡슐화
4. 상속
5. 다형성

위 5가지를 사용하는 것이 이번 과제의 목표라고 생각합니다.

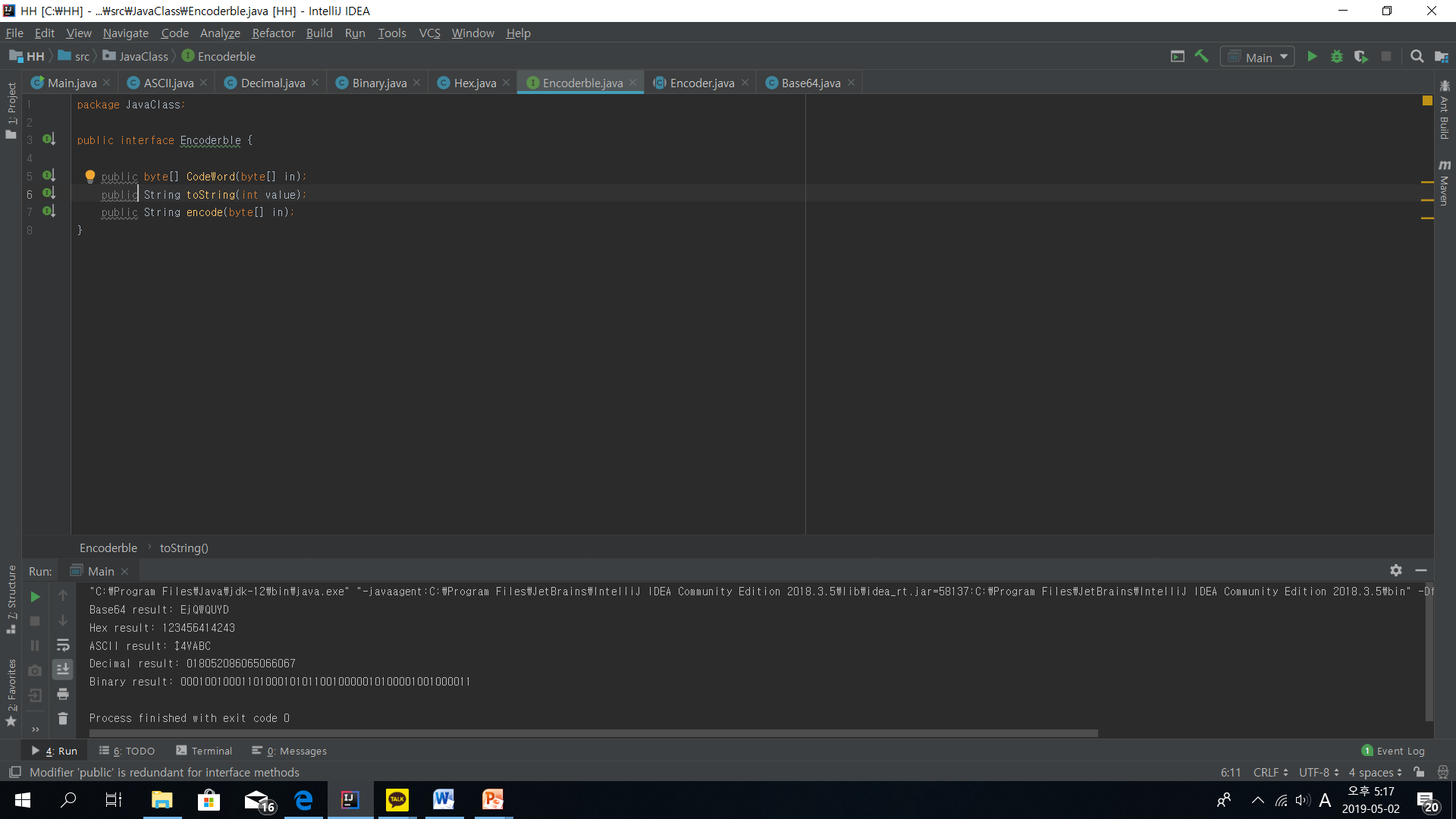
다형성은 오바라이딩(Overriding) 즉 부모클래스의 메서드와 매게변수를 재정의하여 사용하는 것을 말합니다.

따라서 전에 했던 Base64 Encoding 과제를 구조를 나누어 인터페이스 , 부모클래스(또는 추상클래스)를 만들어 상속을 시켜 만들고 상속받은 자식 클래스는 오버라이딩을 통하여 재정의하여 프로그래밍 하는 과제입니다. 그리고 나머지 ASCII, Hex, Decimal ,Binary도 추가하여 같이 상속을 받아 작성하는 과제입니다.

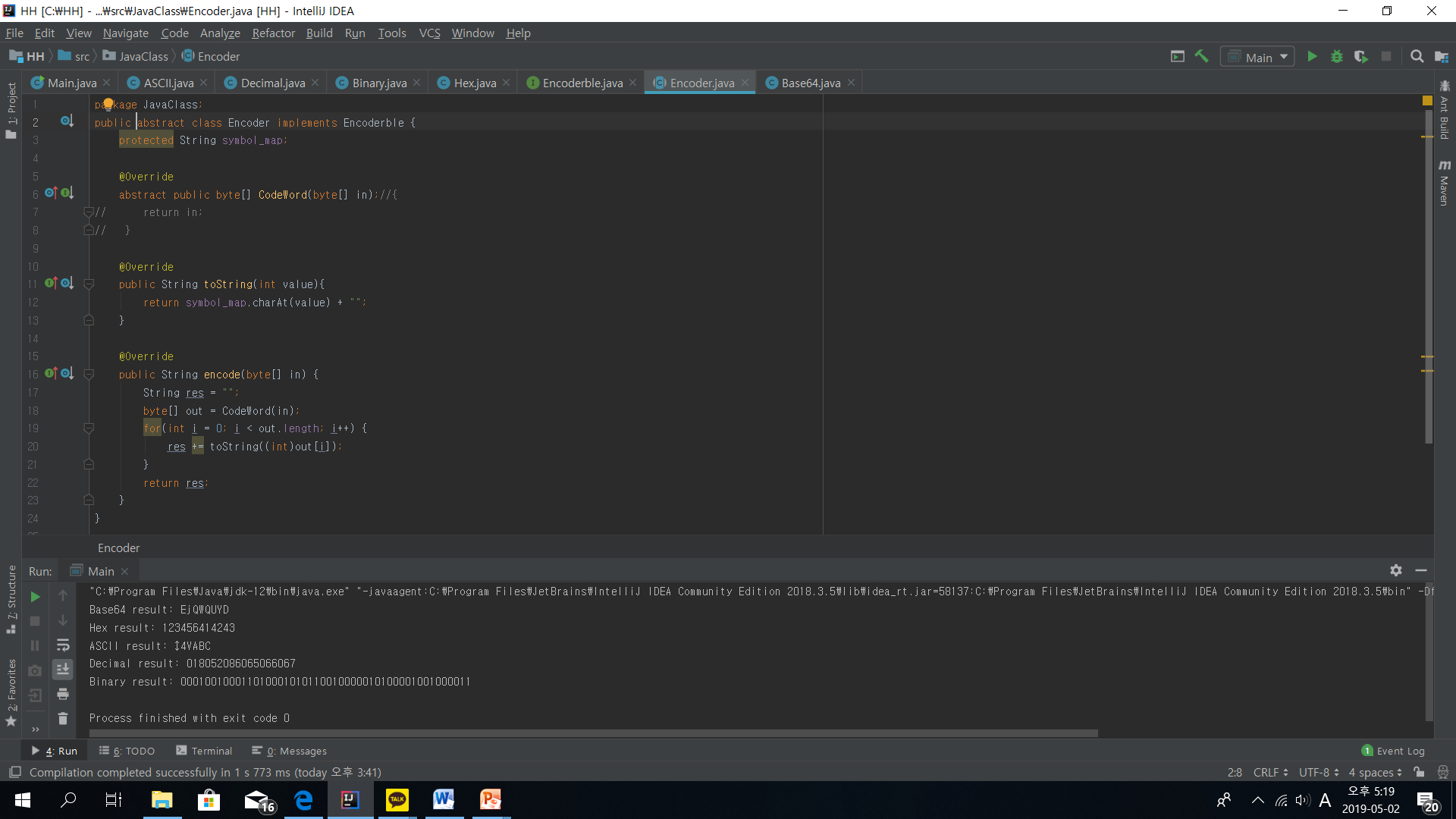
2 Project Goals

이번 과제는 첫번째 과제였던 Base64 Encoding과제를 좀더 확장하여 ASCII, Base64, Binary, Decimal, Hex 5가지로 확대하고 객체지향적인 프로그래밍 작성을 위하여 Hierarchy와 Interface 그리고 Abstract 구조를 사용하여 코드를 간결하게 줄이고 객체지향적인 프로그래밍 작성법을 공부할 수 있도록 하는 과제입니다.

3 Concepts used in Hierarchy, Interface, Abstract + 프로그램 구조

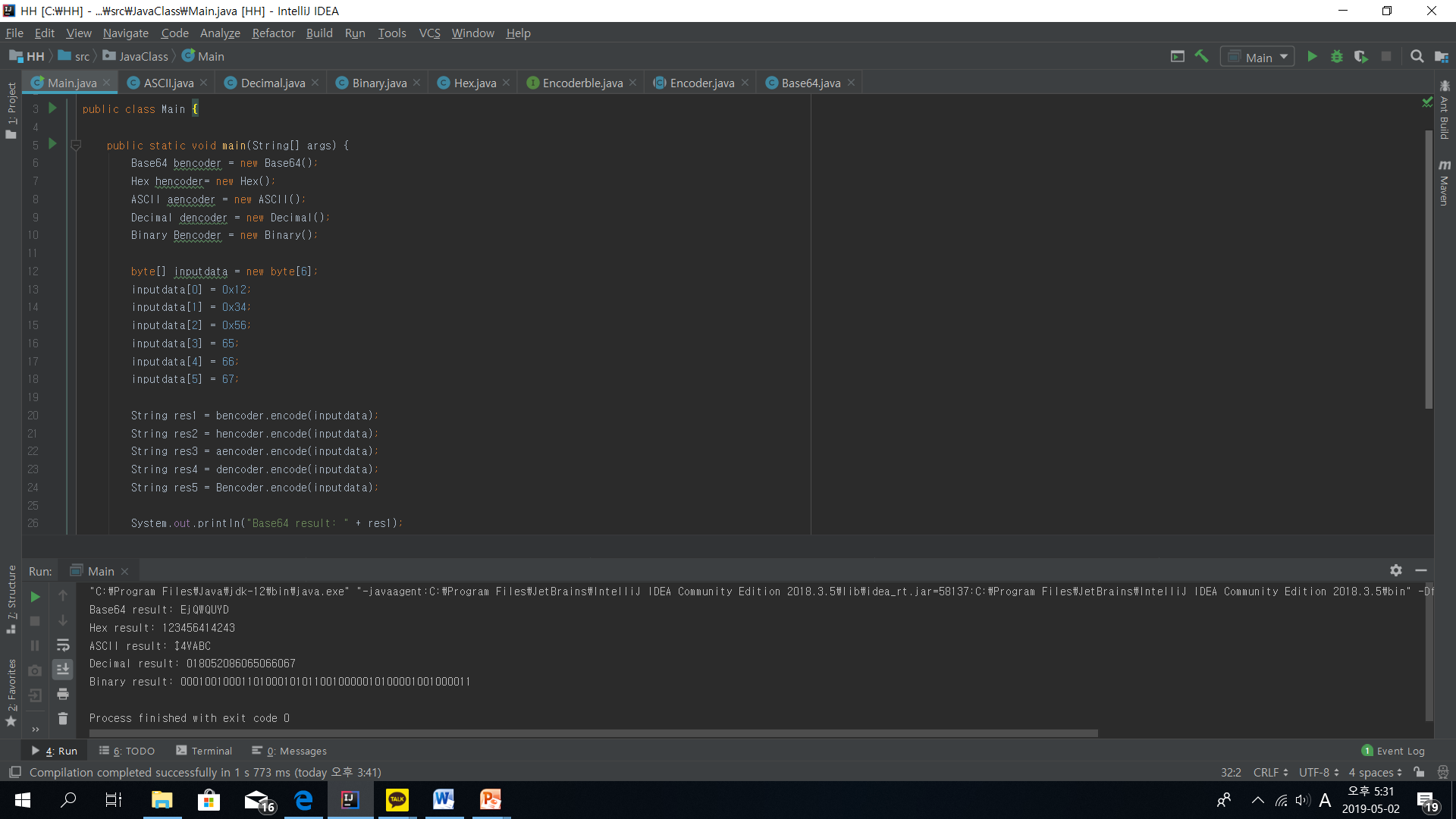
가장 먼저 작성한 것은 이러닝에 올라온 Base64 코드를 기준으로 먼저 Hierarchy와 Interface를 작성하여 상속받는 것 입니다. 

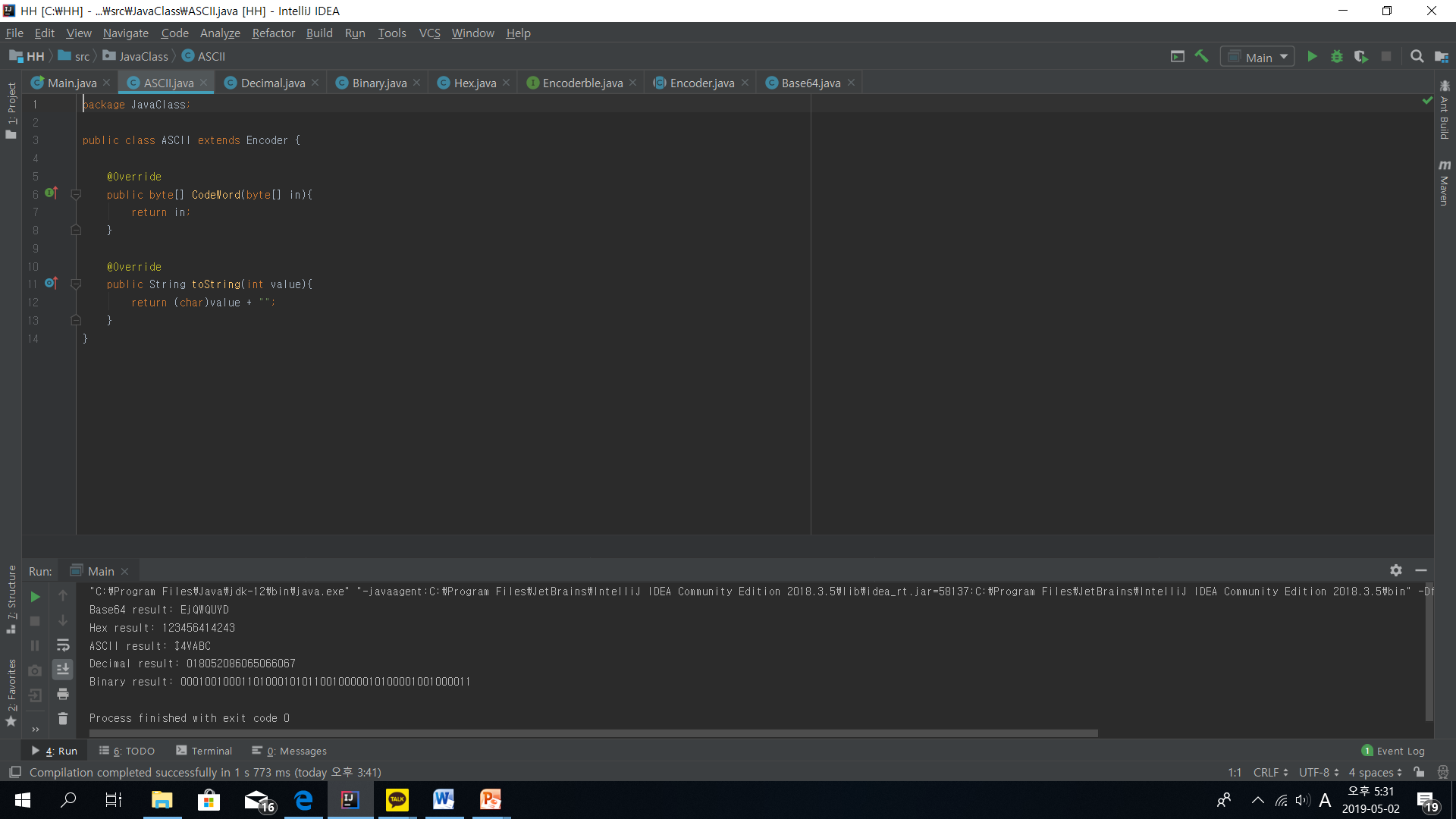
인터페이스인 Encoderble입니다. 부모클래스인 Encoder에서 공통으로 사용하는 메소드인 3개 CodeWord와 toString, encode를 작성하여 이 인터페이스를 부모클래스인 Encoder가 상속받도록 작성하였습니다.



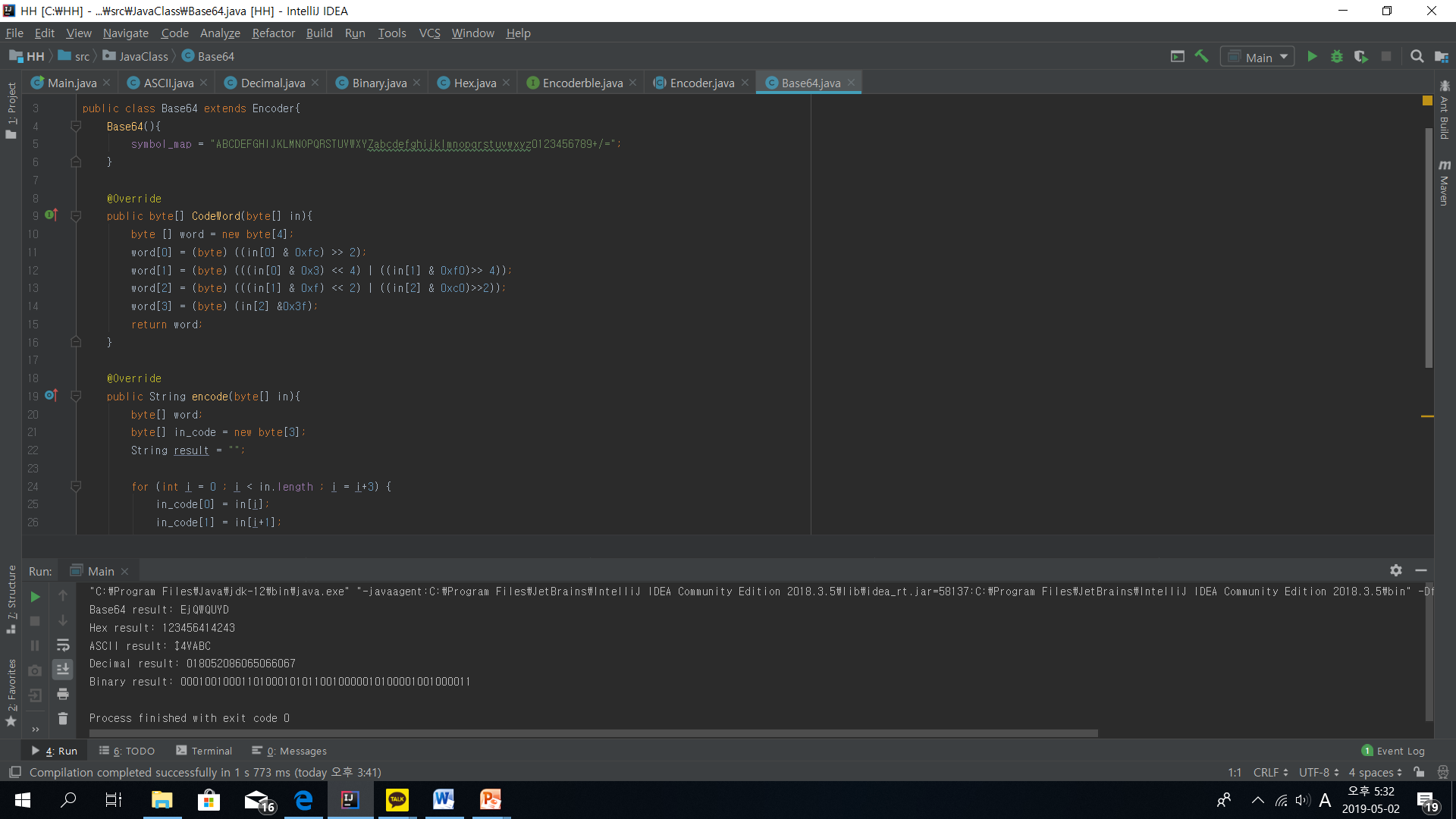
결과 화면은 abstract로 되어있는데 추상클래스가 아닌 그냥 단순 부모클래스는 abstract를 제거하고 주석부분을 다시 원상태로 돌려놓으면 부모클래스가 됩니다. Abstract 클래스를 작성할 때는 선언부만 작성하고 구현부는 작성하지 않은 상태로 내버려 두면 된다. 그러면 이 추상클래스를 상속받는 자식클래스가 오버라이딩을 통해 구현하면 된다. 추상메소드는 반드시 상속받는 자식클래스에서 모두 구현되어야 하며 위 코드에서는 abstract를 붙이지 않은 나머지 메소드와 변수들은 모두 공통으로 사용되고 변형이 되지 않기에 abstract로 바꾸지 않았으며 변형되는 CodeWord만 abstract로 사용하였습니다. 그리고 자식클래스로 ASCII, Base64, Binary, Decimal, Hex 5개의 자식클래스를 만들고 이를 실행할 Main클래스를 따로 작성하여 실행하도록 작성하였습니다.

메인 클래스

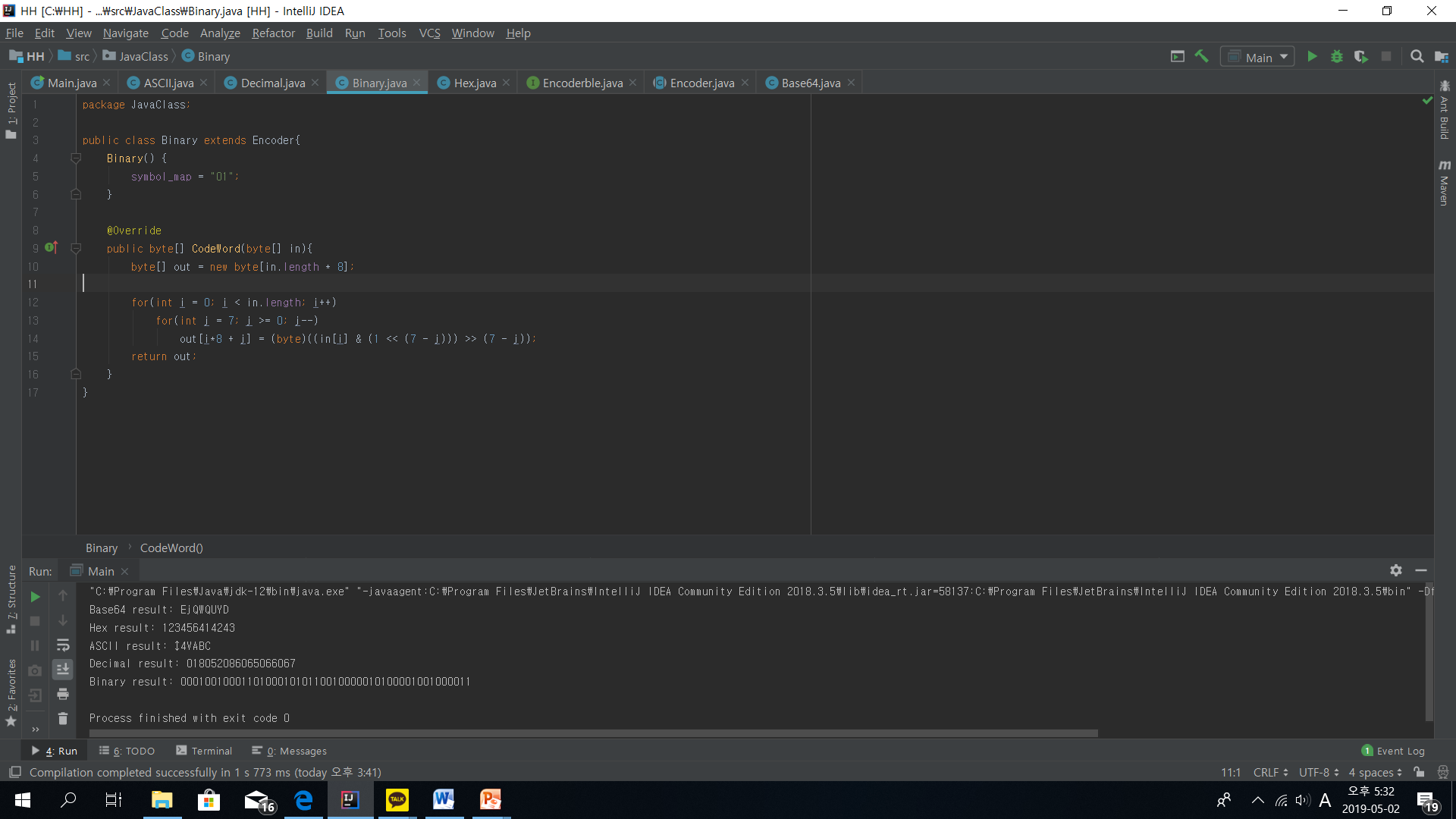


ASCII 클래스

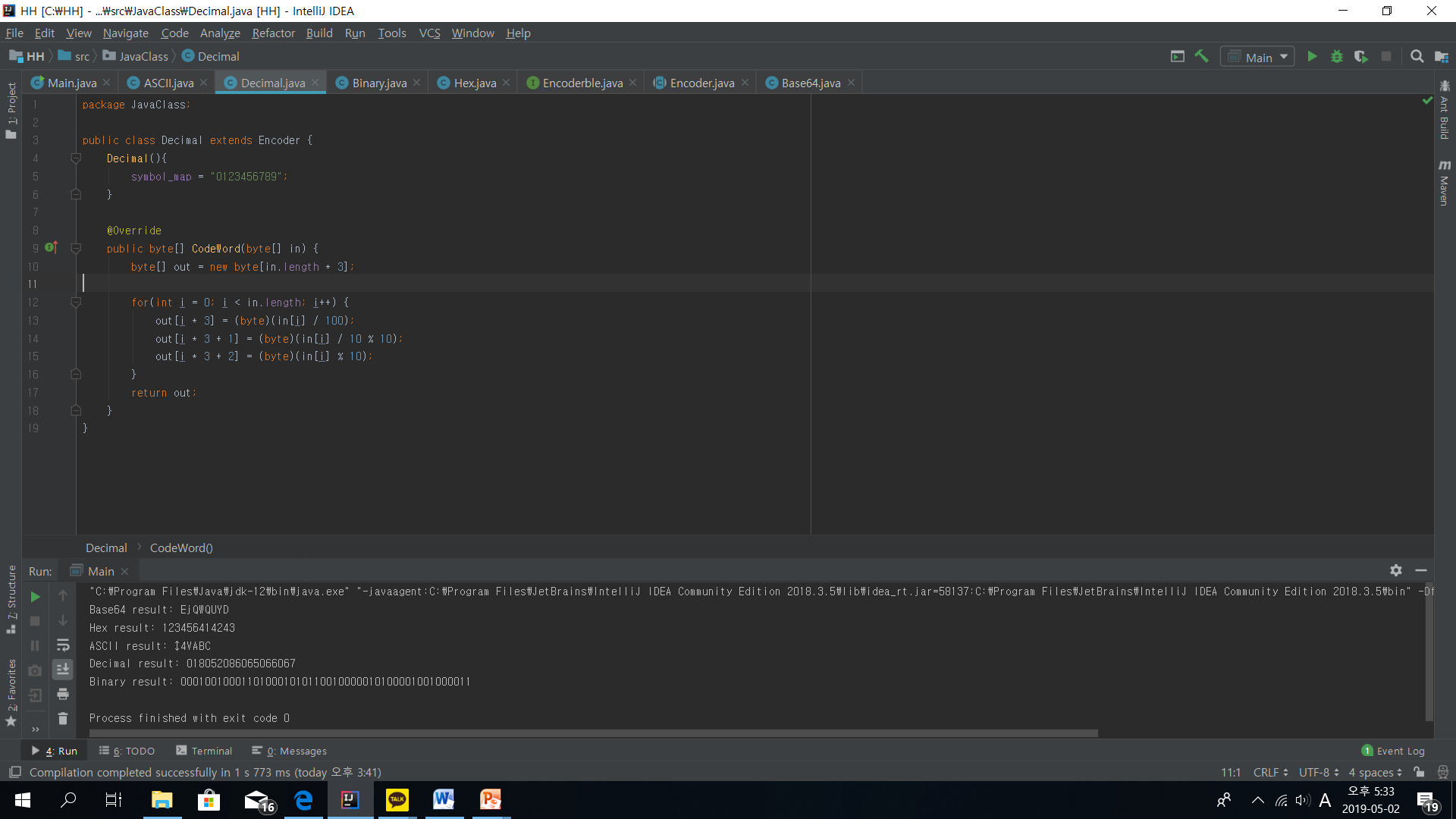
Base64클래스



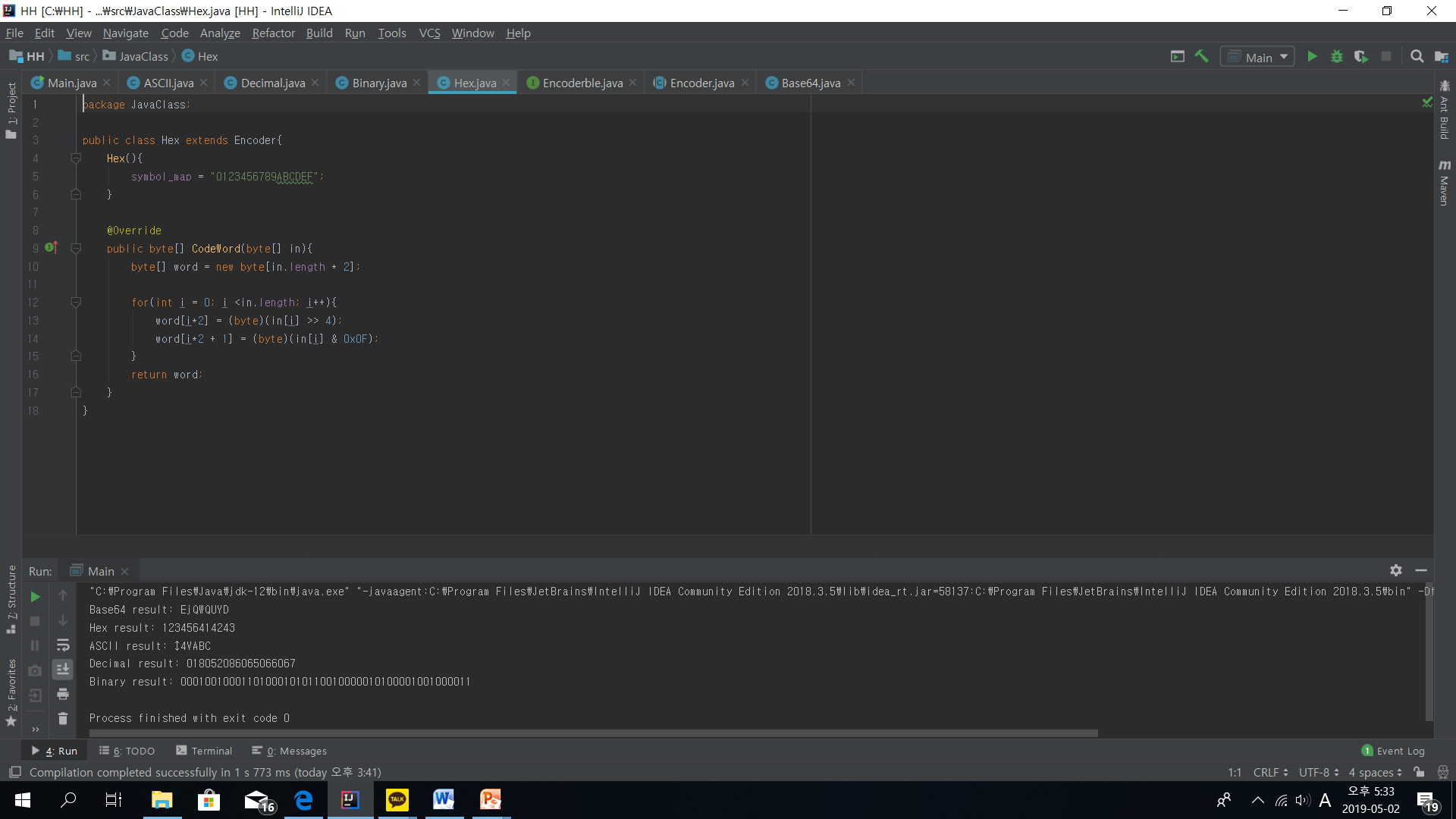
Binary클래스



Decimal 클래스



Hex클래스



인터페이스인 Encoderble과 Hierarchy(+ Abstract)는 위에 있습니다.

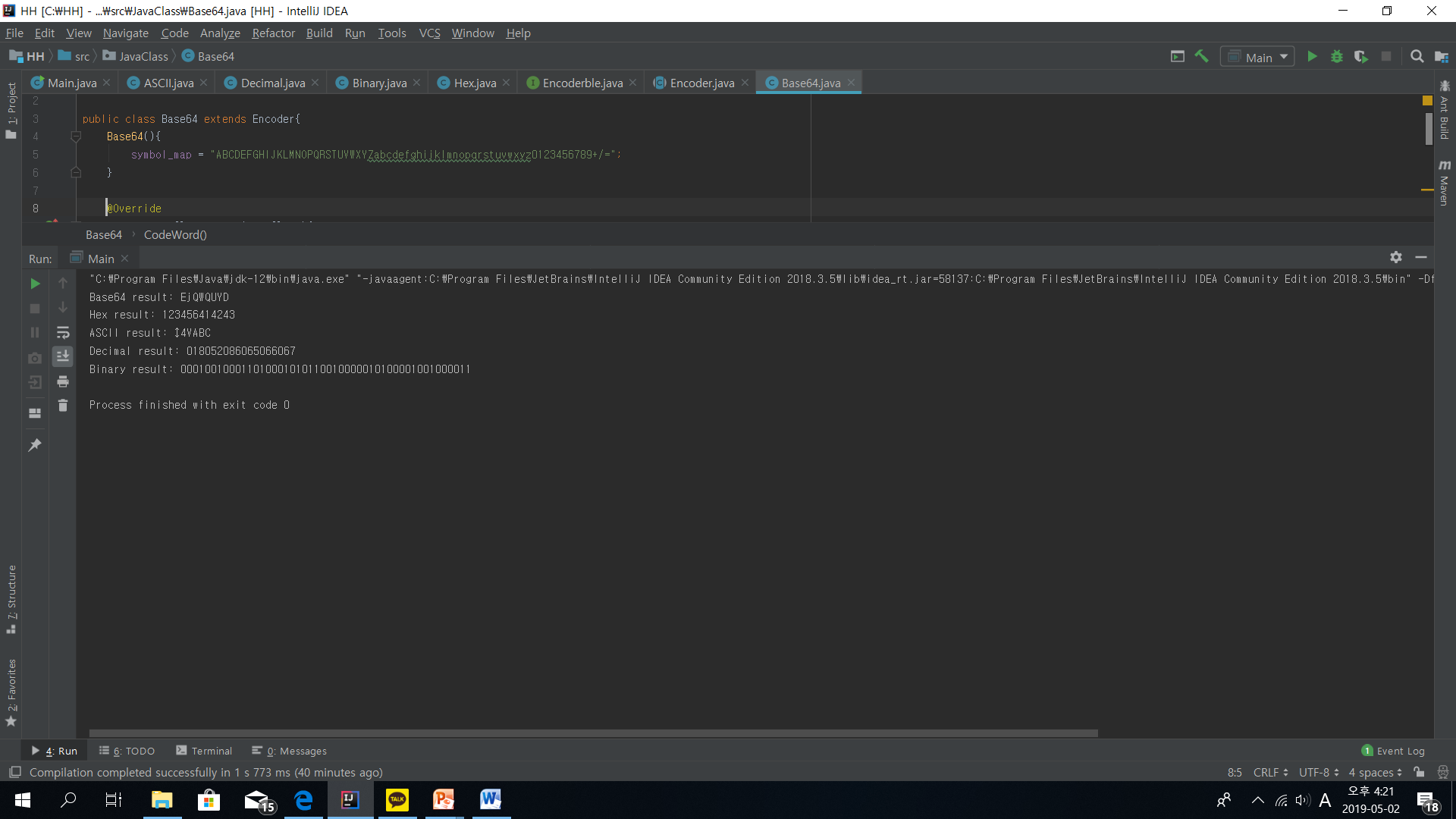
4 Problems and solutions

첫번째 문제는 Hierarchy인 부모클래스 Encoder와 Interface인 Encoderble을 어떻게 사용하는가 였습니다. 수업때 봤었던 교수님의 코드는 기억상으로는 둘다 extends와 implements를 한꺼번에 각 Ascii, Base64, Binary, Decimal, Hex에 받아와 사용하였는데 제가 작성할때는 Base64부터 작성하여 늘려가는데 Base64처음부터 오류가 나왔습니다. 해결방안을 떠올리며 계속 구글링을 해보다가 한 페이지에서 인터페이스를 사용하는 이유는 나중에 여러명이서 공동으로 작업을 하게 될 때 전체적인 구조를 작성하는 사람과 그 밑의 클래스를 담당하는 사람이 따로 작업을 할 경우에 인터페이스나 추상클래스를 먼저 작성하여 그 구조에 맞춰서 작성해 나중에 서로 코드를 합칠 때 합의된 사항으로 코드를 작성하였기에 좀더 수월하게 합칠 수 있다는 글을 보았습니다. 그래서 방식을 바꾸어 부모 클래스인 Encoder에 인터페이스인 Encoderble을 implements하여 인터페이스에서 규정하는 함수들을 모두 사용하고 수정이 필요한 부분은 Override하여 수정한 후에 Encoder를 Base64에다 상속 시키니 오류가 뜨지 않고 Base64의 변환값을 얻을 수 있었습니다. 그렇게 하여 나머지 ASCII, Hex , Decimal , Binary도 작성을 하여 값을 얻을 수 있었습니다. 두번째 문제는 CodeWord부분이였는데 ASCII를 어떻게 나눠야 인코딩을 할수 있는지가 문제였습니다. 계속 못하고 특수 문자도 있고 문자 체계도 달라 처음에는 출력할 수 없는 특수문자를 제외한 32번째부터 126번쨰까지만 출력하도록 하였으니 교수님의 답안에 나왔던 맨 처음의 특수문자 하나가 출력되질 않아 고민하다가 선우의 도움을 받아서 교수님이 수업시간때 작성하셨던 사진을 받아서 작성하여 출력을 하였습니다.

5 Build environment Compilation:

집 컴퓨터에 Eclipse가 설치되어 있어서 원래 Eclipse로 작업하고 있었는데 컴퓨터가 고장나 노트북으로 작업하였습니다. 노트북은 Eclipse가 설치가 되질 않아 IntelliJ만 작동이 되어 IntelliJ로 작성하게 되었습니다. 죄송합니다. 코드는 따로 텍스트 파일로 만들어서 올렸습니다.

6 Screen capture Works well in your IDE screen! Input program, your result



7 Personal feelings

시험도 껴있고 예비군도 갔다오느라 과제 하는데 시간이 부족하여 늦게 제출하게 되었습니다. 죄송합니다. 중간에 컴퓨터도 망가져서 더 늦었습니다..

이번 과제는 객체지향적인 프로그래밍을 만드는 과제로서 프로그래밍 보다는 계층구조 , 인터페이스 , 추상클래스로 나눠서 프로그래밍 하는 과제였는데 개념은 이해하겠는데 어떤 것을 계층 , 인터페이스 ,추상클래스로 나눠야 하는지가 헷갈려 더 오래걸렸습니다.