0721 ABAP OBJECT

OBJECT ORIENTATION |

텍스트, 지도이(가) 표시된 사진

자동 생성된 설명

ABAP OBJECT 가 등장하면서 CLASS | INTERFACE | EVENT 가 등장한다. 이후 커널이 변경되고 상속 등의 개념이 적용되어 객체 지향 | OBJECT-ORIENTED 프로그램이 완성되었다. 그림에서 보듯이 ABAP 언어가 개발되기 전 SAP는 MACRO ASSEMBLER를 이용하여 프로그램을 개발하였다. ABAP 언어는 리포팅 기능을 개선하고자 개발된 프로그램 언어로서, 초기에는 COBOL과 PASCAL 과 같은 영향을 많이 받았다. ABAP OBJECT 는 ABAP의 기능이 확장된 언어이며, JAVA와 C++과 같은 객체지향 OBJECT-ORIENTED 프로그래밍의 장점들이 통합되어 개발되었다. 과거의 정보 시스템은 INPUT과 OUTPUT의 관계로 정의된 기능을 중심으로 이루어졌다. 그러나 객체지향 | OBJECT-ORIENTED 접근법에서는 실제 세계를 구조화 또는 추상화 | ABSTRACT 하여 표현하는 객체에 초점을 맞추고 있다. 이러한 객체들은 구조를 나타내는 속성 | ATTRIBUTE과 기능을 표현하는 METHOD로 구성되어 있다.

실세계의 정보와 상황 중에 주된 관심이 대상이 되는 부분을 컴퓨터 내부로 이식시키는 것이 개발자의 임무이나 실제 세계의 복잡성 때문에 실세계의 상황을 바로 컴퓨터 내부로 반영하지 못한다. 프로그램 개발자는 ‘추상화’와 ‘구체화’ 과정을 통해 그들의 임무를 수행한다. 즉, ‘추상화’ 과정을 통하여 추상적 모델을 소프트웨어 오브젝트로 변환하는 것이다. 아래의 그림과 같이 구체화된 소프트웨어 오브젝트는 프로그래머에게 실세계와 소프트웨어 솔루션을 1:1로 설계할 수 있도록 한다.

스크린샷이(가) 표시된 사진

자동 생성된 설명

프로그램을 개발할 시 구조적인 프로그램 모델과 객체지향 모델을 구별하고 사용할 수 있어야 한다. 기본적으로 두 가지 프로그램 모델 모두 ABAP 프로그램에서 데이터를 선언하고 데이터를 어떻게 처리할 것인가에 대한 문제에 기초를 두고 있다. 절차적인 프로그램에서는 INPUT과 OUTPUT 값을 처리하는 재사용성 | REUSABILITY 기능을 제공하며, 객체지향 프로그램 모델에서는 절차적인 구조에 클래스를 추가하여 상속을 정의하고 메서드를 호출하는 확장된 기능을 지원한다. 클래스의 상속은 여러가지 이점을 가지는데, 그 중 하나는 코드의 재사용이 가능하다는 것이다. 여러 개의 클래스에서 공통으로 사용되는 코드는 그들의 슈퍼클래스 안에 놓일 수 있기 때문에 각각의 새로운 클래스는 굳이 같은 코드로 다시 정의할 필요가 없다. 클래스는 절차적인 프로그램에서는 구현할 수 없는 프로그램을 지원한다. 데이터를 캡슐화 | ENCAPSULATE 하며, 클래스의 메서드를 호출하고, 클래스와 독립적인 객체를 생성할 수 있게 한다. 모든 객체는 하나의 개별 오브젝트로 작동하며, 이러한 오브젝트들의 조합으로 새로운 기능을 창출할 수 있게 된다. 즉, 객체지향 개념은 소프트웨어를 여러 개의 객체의 모임으로 생각하며 객체는 데이터와 관련 함수를 모아 놓은 것을 의미한다. 소프트웨어 개발 시 분석, 설계 그리고 구현에 이르는 모든 부분을 객체에 기반을 두고 있다. 이전의 소프트웨어 개발 방식은 비즈니스 절차 기능 구현에 의존하기 때문에 구조적 혹은 절차적 프로그래밍이라 부른다. 일반적으로 객체지향 프로그래밍을 사용하면 구조적 프로그래밍보다 모듈화가 쉬우며 객체의 여러 가지 특성을 사용하여 소프트웨어 재사용성과 확장성을 높일 수 있다. 그 결과 프로그램 개발 기간을 현저하게 단축할 수 있다. 객체들은 비슷한 속성과 기능을 가질 수 있으며, 하나의 그룹으로 묶일 수 있다. 유사한 객체들끼리 하나의 그룹으로 만드는 것을 클래스라고 하며, 생성된 클래스는 개별 객체들을 생성하는 청사진의 역할을 하게 된다.

절차적인 ABAP 프로그램에서의 객체지향 개념 ? |

ABAP 프로그램은 절차적인 언어에서 파생되었으며 객체지향 개념은 이후부터 포함되었다. SAP는 기업 애플리케이션으로 비즈니스 로직을 순차적으로 처리하며 공통된 기능은 서브루틴 또는 함수를 이용해 구현하기 때문에 실무에서 ALV를 사용하는 것 이외에는 클래스를 이용한 프로그램을 사용할 일이 거의 없었다. 그러나 WEB 환경과 ALV와 같은 진화된 리포트 툴이 도입됨에 따라 객체지향 프로그래밍은 ABAP 프로그램에서 중요한 기술로 주목받고 있다.

절차적인 ABAP 프로그램에서의 객체지향 개념 ? |

ABAP 프로그램은 절차적인 언어에서 파생되었으며 객체지향 개념은 이후부터 포함되었다