
# Introduction: Object Oriented Programming

**Afsara Tasneem Misha**

**Lecturer**

Department of CSE

Daffodil International University

# Today's Contents

- **Structured Language and Object-oriented Language**
- **Why OOP?**
- **Basic concepts of OOP**

# Structured Language
# and
# Object-oriented Language

# Structured / Procedure Programming Methodology (SPM)

# Basic features of SPM

- Emphasis on doing **algorithms**.
- Large Programs are divided into smaller programs known as **Functions**
- Most of the function shares **global data**.
- Data move around the system **globally** from function to function.
- Function **transfers** the data from one form to another.
- Employs **top-down** approach of Programming.

- **Example: C, Pascal, FORTRAN**

# Problems with Structured Programming Methodology (SPM)

- Reach their limit when project becomes too **large**.
- Large program became more **complex**.
- Functions have **unrestricted** access to **global data**.


# OOP - Breaking the Barriers of SPM

# Key Points

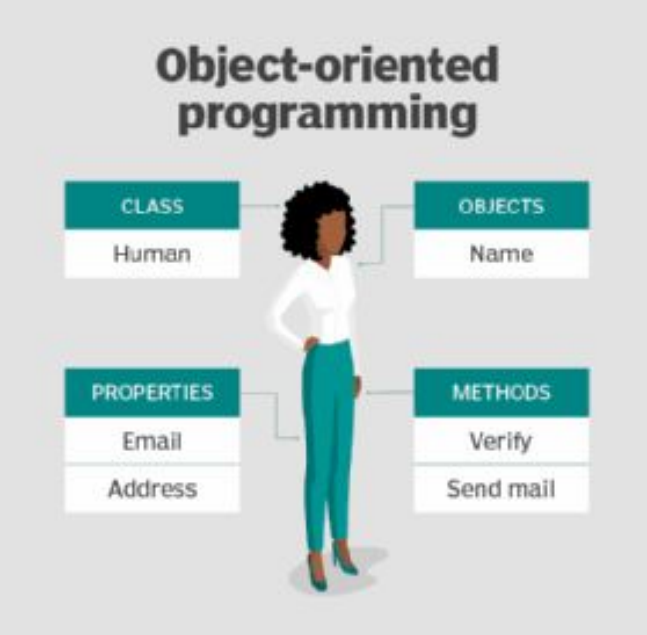
- Took the **best ideas** of **SPM** and combine with several **new concepts**.
- Combine into a single unit both **data** and the **functions** that operate on that data.
- You **can't access** the data **directly**.
- The data is **hidden**, so it is safe from **accidental alteration**.
- Data **encapsulation** and **data hiding** are key terms in the description of object-oriented languages.

# The striking features of OOP

- Emphasis on **data** rather than the **procedure**.
- Programs are divided into **objects**.
- Data are **hidden** and **can't be accessed** by external functions
- Object may **communicate** with each other through **methods** (**functions**).
- New **data** & **functions** can be easily added whenever necessary.

# Concepts of OOP

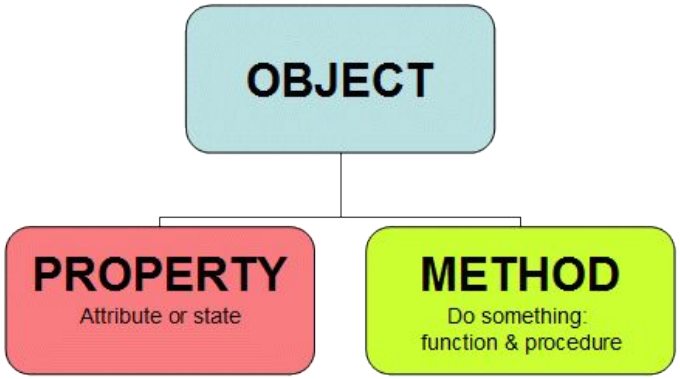
- **Object**
- **Class**
- **Methods**
- **Instance Variables (Properties)**



# 1. Object

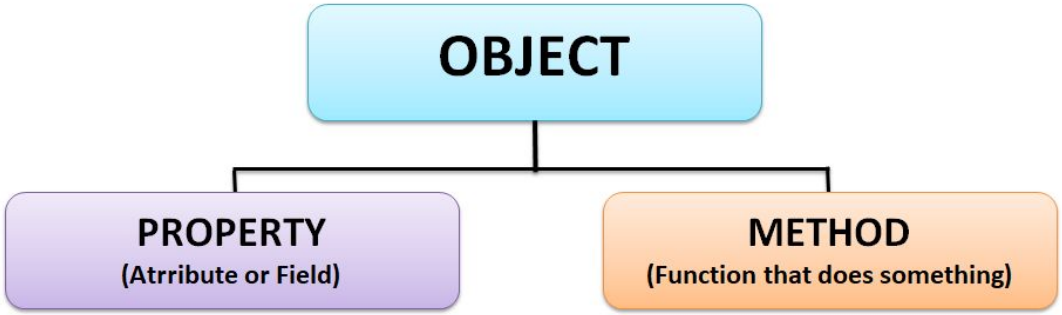
- **Real world entity.**
- Bundle of related **variables** and **functions** (also known **methods**).
- Objects share **two** characteristics:

  1. **Properties / State**
  2. **Method / Behavior (Functionalities)**



# Two characteristics of Object

- Objects share **two** characteristics:




**1. Properties / State**

» State is a well-defined condition of an item.
» A state captures the relevant aspects of an object

**2. Method / Behavior (Functionalities)**

» Behavior is the observable effects of an operation or event

# Example

**Object**: House

**States**: Color

Location

**Behaviors**: Close/open doors

# Try?

**Object**: Car

**States**(Properties): ?

**Behaviors**(Functions): ?

# Try?

**Object**: Car

**States**(Properties): Color, Model#, Wheel

**Behaviors**(Functions): Move, Break

# Try?

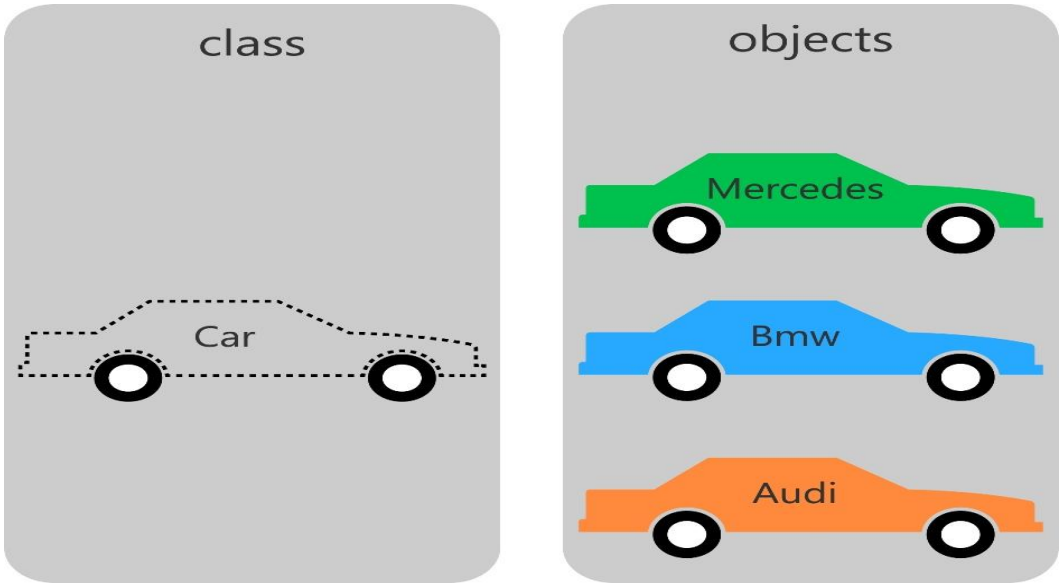
**Object**: ?

**States**(Properties): ?

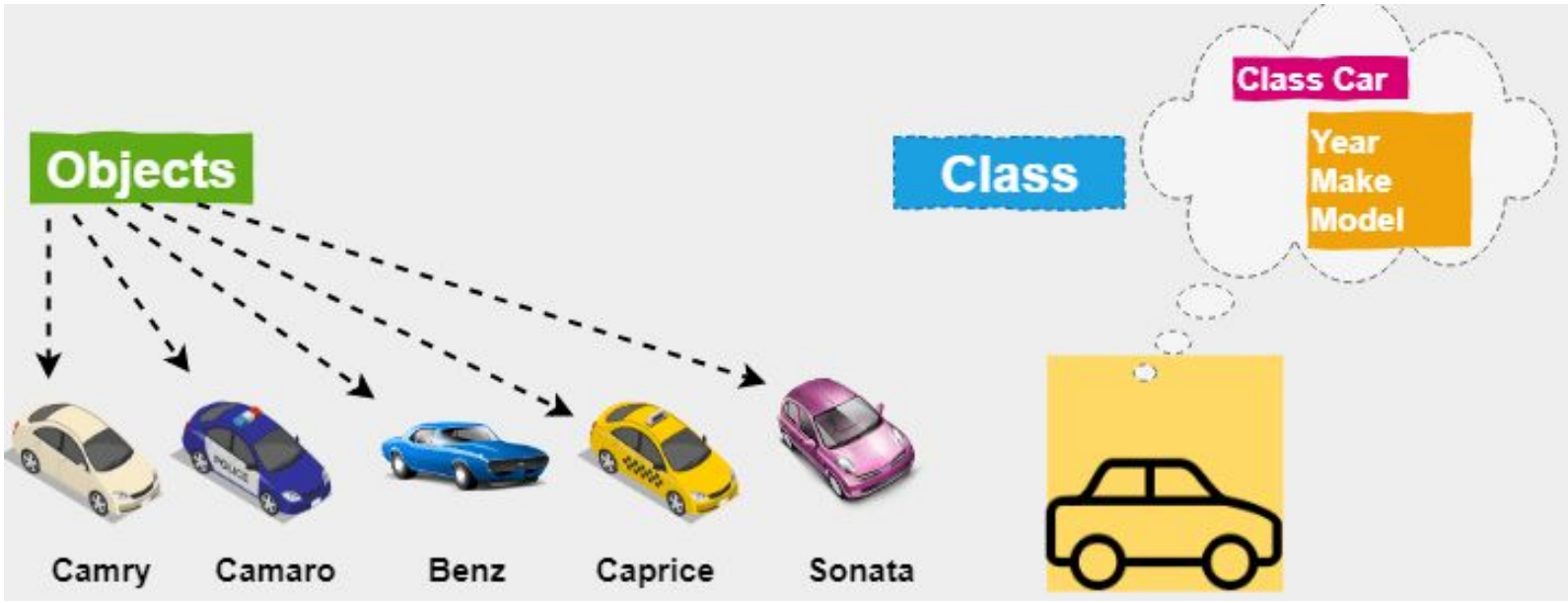
**Behaviors**(Functions): ?

# 2. Class

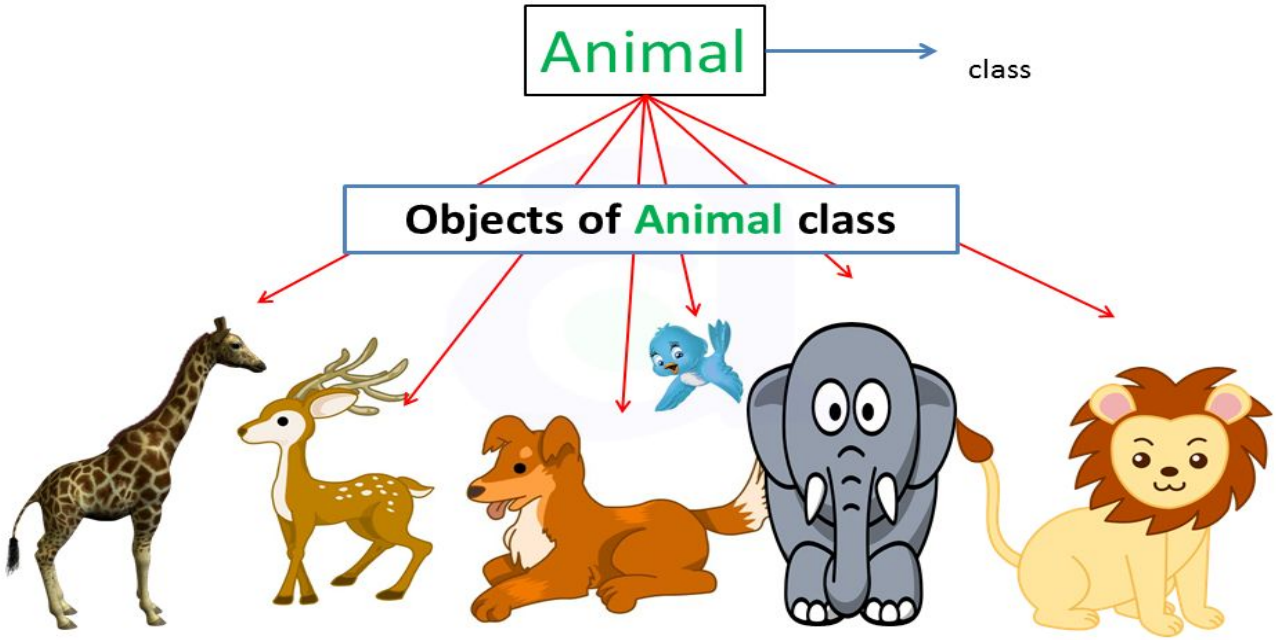
- A class can be defined as a **template/blueprint** of an object that describes the **behaviors/states** that object.



# Class: A template/blueprint of an objects

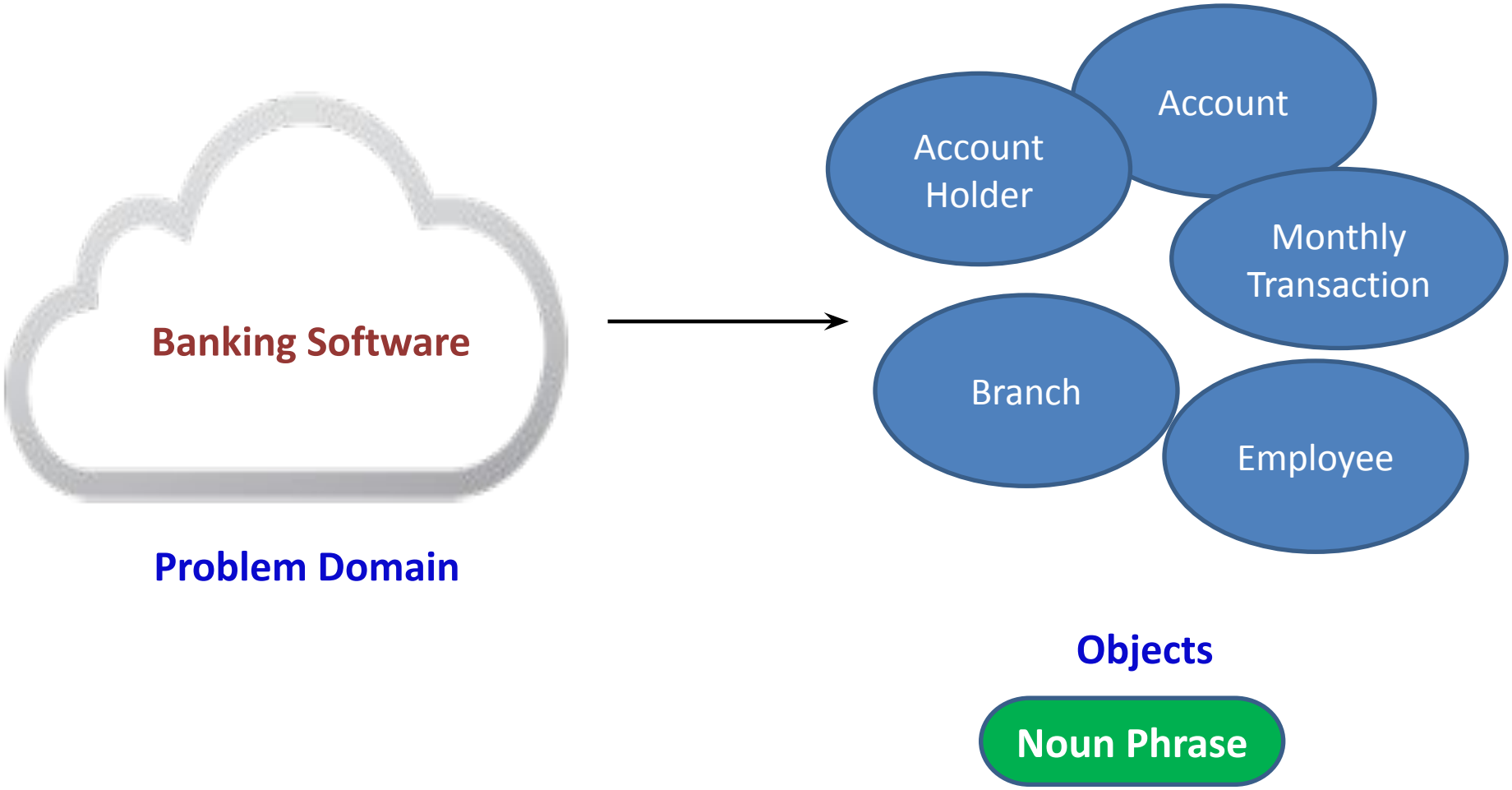


# Class: A template/blueprint of an objects

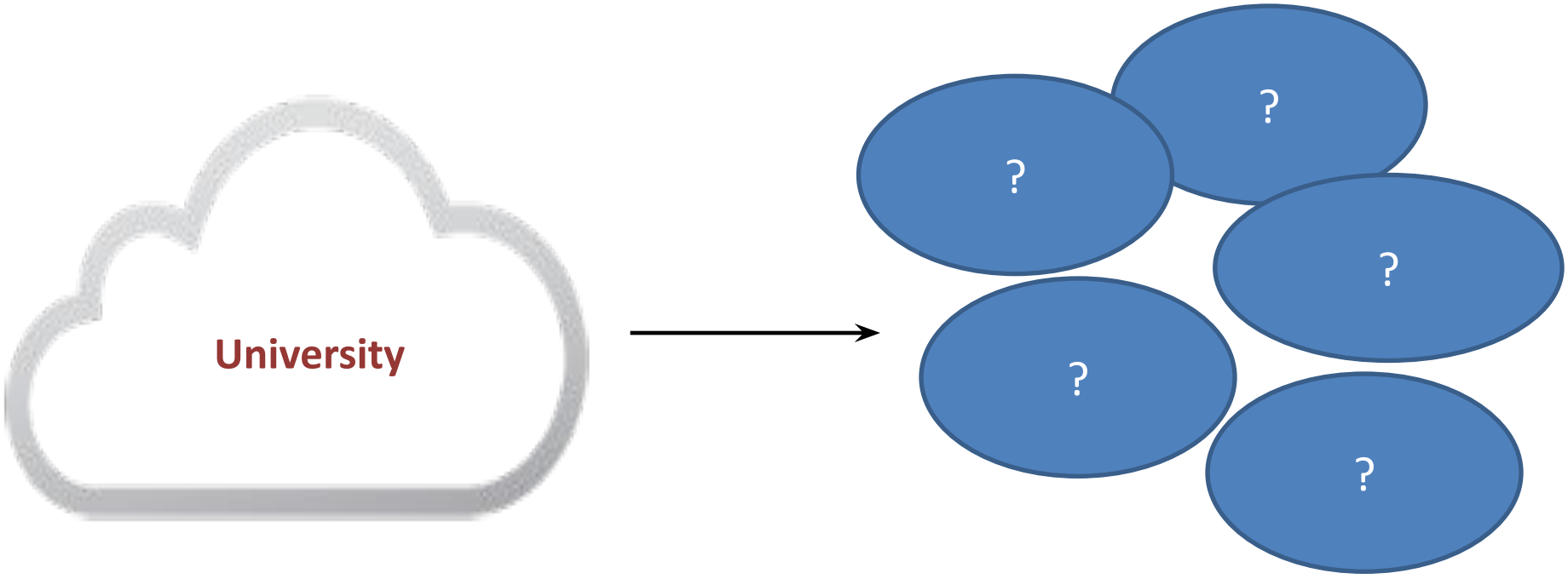


# Object Oriented Analysis

# What is Object Oriented Analysis?



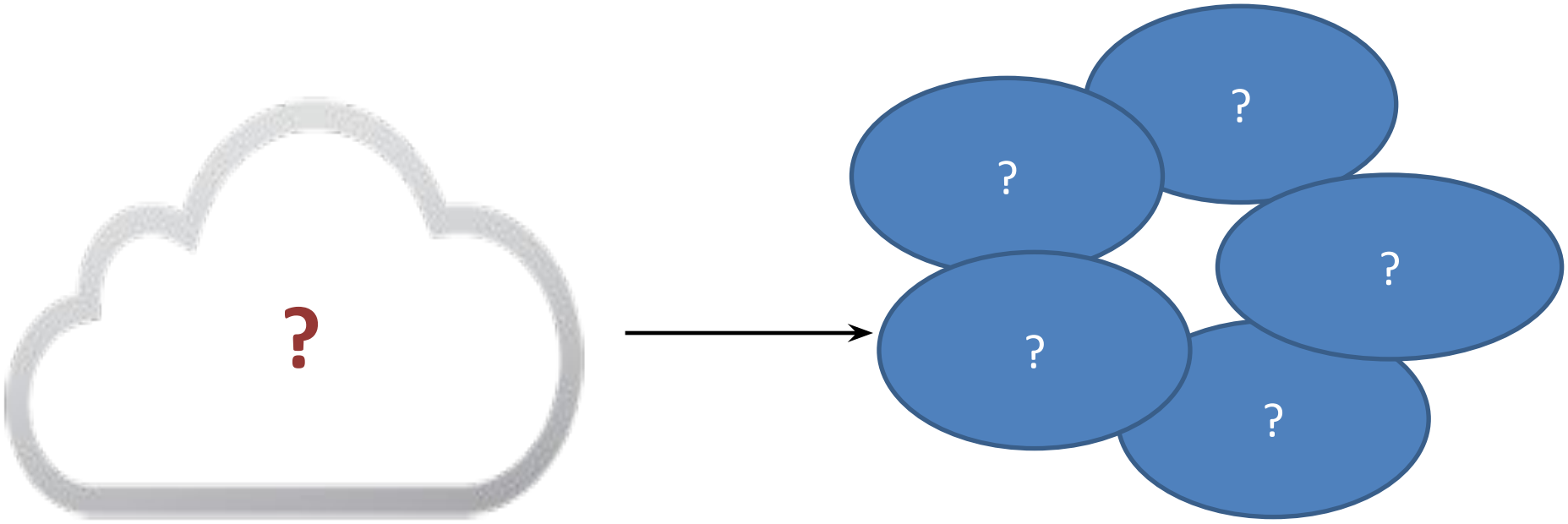
# Have a try




**Problem Domain**

**Objects**

# SelfWork




**Problem Domain**

**Objects**

# Detailing Objects


**Account Holder**

**Account Holder**

Has: name, contact no, accounts

**Account**

**Account**

Has : A/C Number, Balance, Opening Date, Account Holder details

Does:
Deposit money, Withdraw money

**Branch**

**Branch**

Has: Name, Location, No of accounts

Does: Create new accounts, make transaction

**Employee**

**Employee**

Has: name, id, email, salary

Does: Handle banking transactions

# Detailing Objects

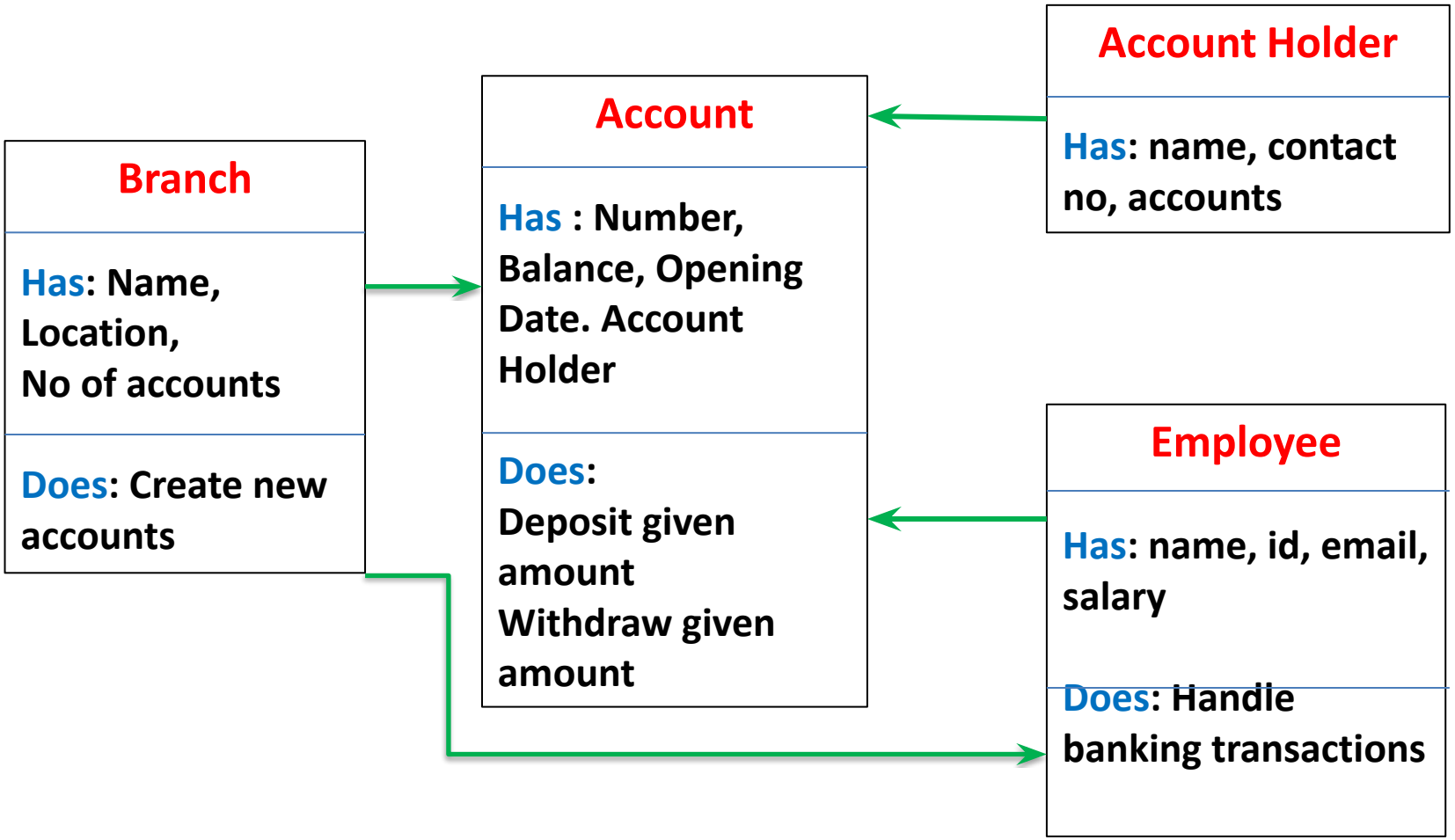

| Branch |
| --- |
| **Attributes**: Name, Location, No of accounts |
| **Function**: Create new accounts |

| Account |
| --- |
| **Attributes** : Number, Balance, Opening Date. Account Holder |
| **Function** : Deposit given amount Withdraw given amount |

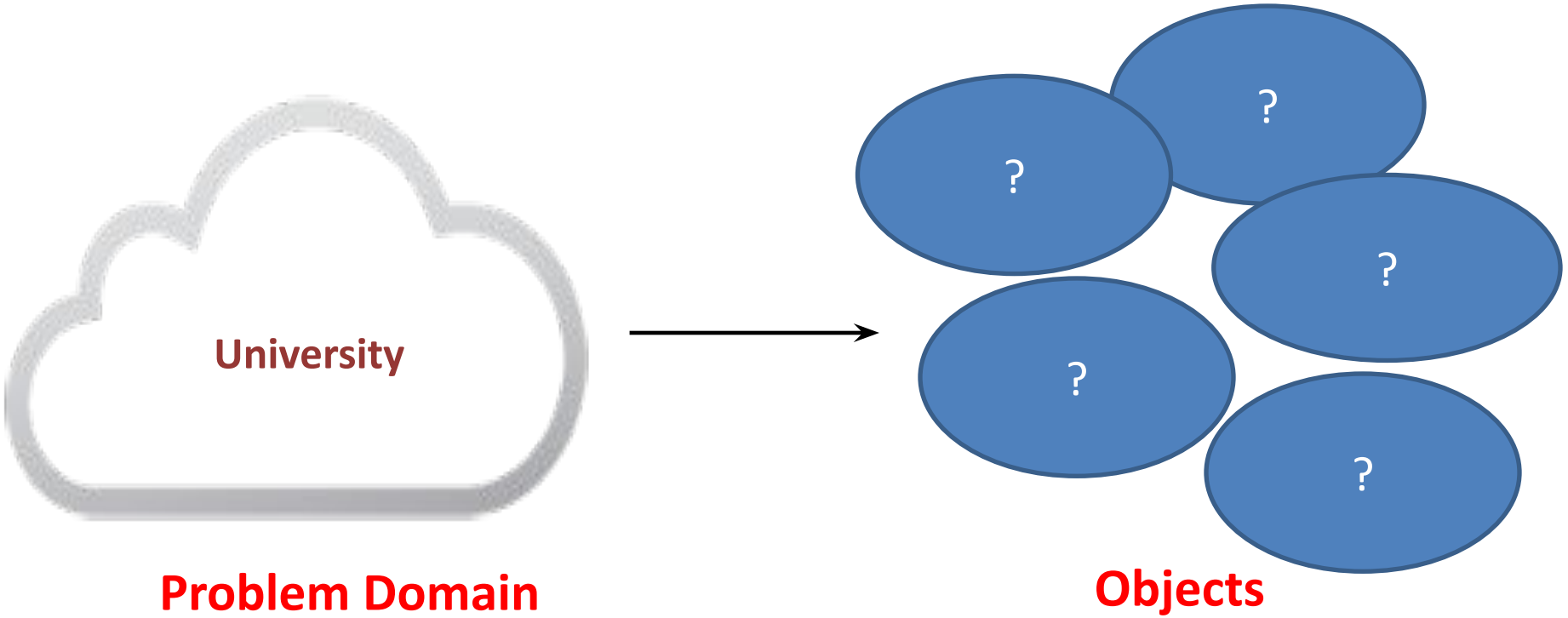
| Account Holder |
| --- |
| **Attributes** : name, contact no, accounts |

| Employee |
| --- |
| **Attributes** : name, id, email, salary |
| **Functions**: Handle banking transactions |

# Objects are inter-connected



# Detail objects of University and Find their inter-connection




Problem Domain

Objects

*Thank you!*