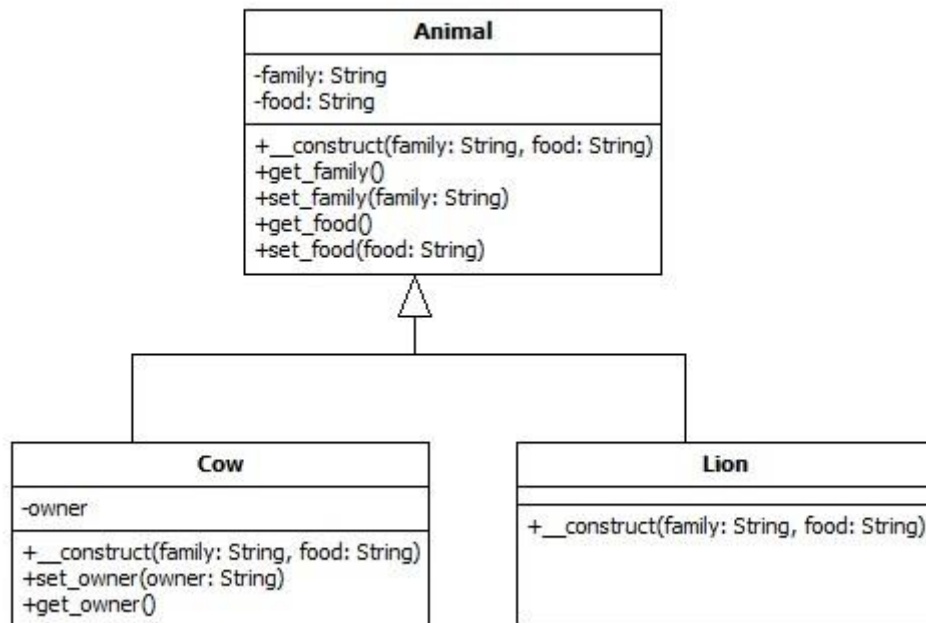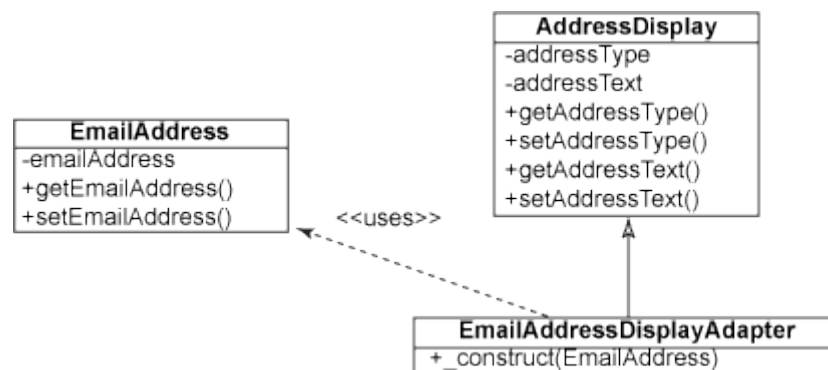# Design pattern in PHP (Lab Assignment 2)

1. Understand / Follow class diagram and Implement logic using PHP OOP approach. (hint : create 4 files, Animal.php, Cow.php, Lion.php, Main.php)

```
                    ┌─────────────────────────────────────────────┐
                    │                   Animal                    │
                    ├─────────────────────────────────────────────┤
                    │ -family: String                             │
                    │ -food: String                               │
                    ├─────────────────────────────────────────────┤
                    │ +__construct(family: String, food: String)  │
                    │ +get_family()                               │
                    │ +set_family(family: String)                 │
                    │ +get_food()                                 │
                    │ +set_food(food: String)                     │
                    └─────────────────────────────────────────────┘
                                        △
         ┌──────────────────────────────┴───────────────────────────────┐
┌─────────────────────────────────────┐      ┌───────────────────────────────────────────┐
│                 Cow                 │      │                   Lion                    │
├─────────────────────────────────────┤      ├───────────────────────────────────────────┤
│ -owner                              │      │ +__construct(family: String, food: String)│
├─────────────────────────────────────┤      └───────────────────────────────────────────┘
│ +__construct(family: String, food: String) │
│ +set_owner(owner: String)           │
│ +get_owner()                        │
└─────────────────────────────────────┘
```
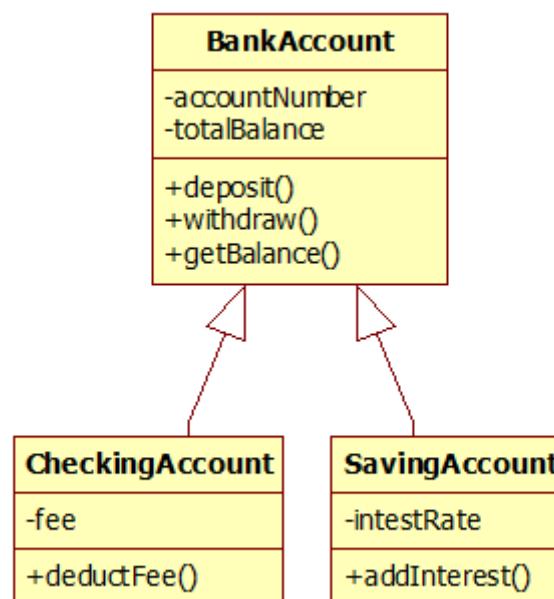
2. Understand / Follow class diagram and Implement logic using PHP OOP approach. (hint : create 2 files classes.php and main.php. create 3 classes AddressDisplay, EmailAddress, EmailAddressDisplayAdapter in classes.php)

AddressDisplay
- -addressType
- -addressText
- +getAddressType()
- +setAddressType()
- +getAddressText()
- +setAddressText()

EmailAddress
- -emailAddress
- +getEmailAddress()
- +setEmailAddress()

<<uses>>

EmailAddressDisplayAdapter
- + construct(EmailAddress)

```php
require_once('classes.php');
$email = new EmailAddress();
$email->setEmailAddress("trueline.chirag@gmail.com");
$address = new EmailAddressDisplayAdapter($email);
echo($address->getAddressType() . "\n") ;
echo($address->getAddressText());
```

3. Understand / Follow class diagram and Implement logic using PHP OOP approach.

BankAccount
- -accountNumber
- -totalBalance

- +deposit()
- +withdraw()
- +getBalance()

CheckingAccount
- -fee

- +deductFee()

SavingAccount
- -intestRate

- +addInterest()

4. Understand / Follow class diagram and Implement logic using PHP OOP approach.

```
                        ┌─────────────────────────────────┐
                        │            Person               │
                        ├─────────────────────────────────┤
                        │ -name:String                    │
                        │ -address:String                 │
                        ├─────────────────────────────────┤
                        │ +Person(name:String,address:String) │
                        │ +getName():String               │
                        │ +getAddress():String            │
                        │ +setAddress(address:String):void│         ┌──────────────────┐
                        │ +toString():String  •-----------│---------│ "name(address)"  │
                        └─────────────────────────────────┘         └──────────────────┘
                                        △
                        ┌───────────────┴───────────────┐
┌─────────────────────────────┐              ┌─────────────────────────────────┐
│          Student            │              │            Teacher              │
├─────────────────────────────┤              ├─────────────────────────────────┤
│ -numCourses:int = 0         │              │ -numCourses:int = 0             │
│ -courses:String[] = {}      │              │ -courses:String[] = {}          │
│ -grades:int[] = {}          │              ├─────────────────────────────────┤
├─────────────────────────────┤              │ +Teacher(name:String,           │
│ +Student(name:String,       │              │    address:String)              │
│    address:String)          │              │ +toString():String              │
│ +toString():String          │              │ +addCourse(course:String):boolean •----┐
│ +addCourseGrade(course:String, │           │ +removeCourse(course:String):boolean •--│--┐
│    grade:int):void          │              │ +toString():String              │   │  │
│ +printGrades():void         │              └─────────────────────────────────┘   │  │
│ +getAverageGrade():double   │                                                    │  │
│ +toString():String •        │         ┌────────────────────────────────────────┐ │  │
└─────────────────────────────┘         │ Return false if the course already existed │◄┘  │
                 ┌──────────────────────┐└────────────────────────────────────────┘    │
                 │ "Student: name(address)" │  ┌──────────────────────────────────────┐    │
                 └──────────────────────┘  │ Return false if the course does not exist │◄───┘
                                           └──────────────────────────────────────┘
                              ┌──────────────────────┐
                              │ "Teacher: name(address)" │
                              └──────────────────────┘
```