ASSIGNMENT – 2

QUESTION – 1

1. Which SOLID principle does the Customer class violate?
   Answer: Customer class violates the Single Responsibility Principle.
2. Why does the code violate the principle?
   Answer: As we can see in the code the Customer class is handling multiple responsibility like figuring out the whether the customer is of specific nation and specific state and also will send an email of customer offer. As a result, it is violating Single Responsibility Principle.
3. Write the code that fixes the violation.
   Answer: The code with the violation fixes is available in the Q1 folder.

QUESTION – 2

1. Which SOLID principle does the USDollarAccount class violate?
   Answer: USDollarAccount class violates the Liskov Substitution Principle.
2. Why does the code violate the principle?
   Answer: USDollarAccount extends the Account class and does changes the way credit and debit method works, but does not changes the balance method, as a result the credit and debit had exchange rates applied to it, but while returning the balance the exchange rates were not applied as a result the sub class breaks the super class's API contract by not providing the actual balance data. As a result, it violates the Liskov Substitution Principle.
3. Write the code that fixes the violation.
   Answer: The code with the violation fixes is available in the Q2 folder.

QUESTION – 3

1. Which SOLID principle does the Student class violate?
   Answer: Student class violates the Single Responsibility Principle.
2. Why does the code violate the principle?
   Answer: Student class is handling multiple responsibility like saving the Student data to a file and also loading it from the file. As a result, Single Responsibility Principle is violated.
3. Write the code that fixes the violation.
   Answer: The code with the violation fixes is available in the Q3 folder.

QUESTION – 4

1. Which SOLID principle does the Employer class violate?
   Answer: Employer class violates the Dependency Inversion Principle.
2. Why does the code violate the principle?
   Answer: Employer class is directly dependent on the HourlyWorker and SalaryWorker classes rather than interacting with their abstraction, as a result any changes in HourlyWorker and SalaryWorker code might also need a change in Employer class which is not good. Employer class must depend on the abstraction of the HourlyWorker and SalaryWorker. As a result, Dependency Inversion Principle is violated.
3. Write the code that fixes the violation.
   Answer: The code with the violation fixes is available in the Q4 folder.

QUESTION – 5

1. Which SOLID principle does the following code violate?
   Answer: The following code violates the Interface Segregation Principle.
2. Why does the code violate the principle?
   Answer: The IInsect interface contains multiple function which may not be used by the class that inherits it, like as we can see AqauticInsect does not have a function body for fly function and FlyingInsect does not have a function body for swim function. The interface needs to contain only functions that the class inheriting it will have function body for. As a result, Interface Segregation Principle is violated.
3. Write the code that fixes the violation.
   Answer: The code with the violation fixes is available in the Q5 folder.

QUESTION – 6

1. Which SOLID principle does the following code violate?
   Answer: The following code violates the Open/Closed Principle.
2. Why does the code violate the principle?
   Answer: The code violates the Open/Closed Principle because we can see the CountryGDPReport class is printing the GDP of Canada and Mexico, if the Canada and Mexico gets more function for calculating the GDP, CountryGDPReport class will have to include that changes too. A separate interface can be created which will allow the Canada and Mexico class to generate their GDP Report and print it, as a result any changes in the Canada and Mexico class can be reflected in the GDP Report method (Open for extension) and whichever class that uses will not be able to change it (Close for modification).
3. Write the code that fixes the violation.
   Answer: The code with the violation fixes is available in the Q6 folder.

QUESTION – 7

1. Which SOLID principle does the following code violate?
   Answer: The following code violates the Interface Segregation Principle.
2. Why does the code violate the principle?
   Answer: The ILibraryItem interface contains multiple function which is not been used by the class that's inheriting it. As we can see in the Book and DVD class many functions of the ILibraryItem is empty. Separate interfaces can be created, such that the functions inside the interface will be containing function body when inherited by a class. As a result, the code violates the Interface Segregation Principle.
3. Write the code that fixes the violation.
   Answer: The code with the violation fixes is available in the Q7 folder.