**UML Diagrams (Class Relationships) - Banking Application - HW05**

**Account**

- **Customer** customer

- **double** balance

- **int** account\_number

- **transaction** \* transactions

+ **double** get\_balance()

+ **Customer** get\_customer()

+ **string** to\_string()

+ **void** set\_cutomer()

+ **void** set\_ballance()

**Savings\_Account**

+ **void** deposit()

+ **void** withdraw()

+ **void** add\_interest()

**Transaction**

**- int** customer\_number

- **string** transaction\_type

- **double** amount

- **double** fees

+ **void** proccess\_transaction()

**Bank**

**- vector<Account \*>** account

- **vector<Customer \*>** customers

+ **void** add\_account()

+ **void** make\_deposit

+ **void** make\_withdrawal()

+ **Account** get\_account()

**Adult**

- **const int**  SAVINGS\_INTEREST

- **const int**  CHECK\_INTEREST

- **const int**  CHECK\_CHARGE

- **const int** OVERDRAFT\_PENALTY

**Student**

- **const int**  SAVINGS\_INTEREST

- **const int**  CHECK\_INTEREST

- **const int**  CHECK\_CHARGE

- **const int** OVERDRAFT\_PENALTY

**Senior**

- **const int**  SAVINGS\_INTEREST

- **const int**  CHECK\_INTEREST

- **const int**  CHECK\_CHARGE

- **const int** OVERDRAFT\_PENALTY

**Customer**

- **string** name

- **string** address

- **int** age

- **string** telephone\_number

- **int** customer\_number

+ (accessor and mutators for all data fields)

**Checking\_Account**

+ **void** deposit()

+ **void** withdraw()

+ **void** add\_interest()