**«datatype»**
**Bank**

+createBankAccount(in String accountHolderName: String, in accountNumber: String) {IllegalBankingException}
+getBalanceByHolder(in accountHolderName: String): Real
+addBalance(in accoungNumber: String, in amout: Real) {IllegalBankingException}

**«datatype»**
**IllegalBankingException**

IllegalBankingException(message: String)

**«datatype»**
**Exception**

Exception(message: String)

{readOnly}

-bankAccounts
[0...*]

**«datatype»**
**BankAccount**

-accountHolderName: String
-accountNumber: String
-openingDate: java.time.LocalDate = java.time.LocalDate.now() {readonly}
-balance: Real = 0

+BankAccount(accountHolderName: String, accountNumber: String) {java.lang.IllegalArgumentException}
-checkAccountHolderName(in accountHolderName: String): Boolean
-checkAccountNumber(in accountNumber: String): Boolean
+setAccountHolderName(in accountHolderName: String) {java.lang.IllegalArgumentException}
+addBalance(in value: Real)
-setAccountNumber(in accountNumber: String) {java.lang.IllegalArgumentException}
+getAccountHolderName(): String
+getAccountNumber(): String
+getBalance(): Real
+getOpeningDate(): LocalDate
+toString(): String
+equals(in o: java.lang.Object): Boolean