```
                                              ┌─────────────────────────────┐
                                              │        <<interface>>        │
                                              │   Comparable<Employee>      │
                                              ├─────────────────────────────┤
                                              │                             │
                                              │ +compareTo(Employee: one): int │
                                              │                             │
                                              └─────────────────────────────┘
                                                           │
 ┌──────────────────────────────┐                         │
 │  <<<Ennumeration SortType>>>  │                         │
 ├──────────────────────────────┤                         │
 │                              │                         │
 │ + SORT_BY_LNAME              │                         │
 │ + SORT_BY_ID                 │                         │
 ├──────────────────────────────┤              ┌──────────────────────────────────┐
 │                              │              │            Manager               │
```

(top-left, empty class box)

Employee

empID : int
firstName: String
lastName: String
ssNum: int
hireDate: Date
salary: double

+changeName(last: String, first: String)
+raiseSalary(): double
+Employee(empID: int, firstName: String,
+lastName:String, ssNum: int, hireDate: Date,
salary:double, deptName, String)
+getEmpId():int
+getHireDate(): Date
getSalary(): double
+toString(): String

Manager

deptName: String

+Manager(empID: int, firstName: String,
lastName:String, ssNum: int, hireDate: Date,
salary:double, deptName, String)
+getDeptName() : String
+setDeptName(deptName: String) : String
+toString(): String
+setDept(debt: DeptType)
+getDept(): DeptType
+addEmployee(emp: Employee)
+removeEmployee(emp: Employee)
+List<Employee> getEmpList(): List

Payable

Contractor

- id: integer
- firstName: String
- lastName: String
- ssNum: integer
- hourlyRate: double

Payable

HRUtility

+ empDateFormat: SimpleDateFormat

+ displayEmployee(emp: Employee)
+ displayEmployees(empList; List<Employee>)
+ displayManager(mgr : Manager)
+ strToDate(s: String) : Date
+ dateToStr(d : Date) :String

Account

- balance : double

+ Account(initBalance : double)
+ credit(amount : double)
+ debit(amount : double)
+ processCheck(p : Payable, hoursBilled :
double)

<<interface>>
Payable

calculatePay(double: numHrs): double
getPayTo(): String
getPayMemo() : String

<<Exception>>
InsufficientFundsException