

Consts	
FULLTIME	String
PARTTIME	String
DEFAULTHOURS	int
NAMES	int
ZERO	double
NOTFOUND	int
DEFAULT	int
GROW	int
INVALID	int
QUADRENNIAL	int
CENTURY	int
QUARTERCENTENNIAL	int
ALTMAXDAYS	int
FEBDAYSLEAP	int
FEBDAYS	int
DEFAULTMAXDAYS	int
YEARMIN	int
PAYPERIODS	int
PAYCHECK_MSG	String
SEPARATOR	String
df	DecimalFormat
SALARY_MSG	String
MANAGER_BONUS	double
DEPHEAD_BONUS	double
DIRECTOR_BONUS	double
MA_CODE	int
DH_CODE	int
DI_CODE	int
MANAGER_MSG	String
DH_MSG	String
DIRECTOR_MSG	String
FULLHOURS	int
OVERTIME_RATE	double
HOURLYPAY_MSG	String
HOURS_MSG	String
PARTTIME_MAX	int
SPLITONE	int
SPLITTWO	int
SPLITTTHREE	int
SPLITFOUR	int
SPLITFIVE	int
SPLITSIX	int
ONEINPUT	int
FOURINPUTS	int
FIVEINPUTS	int
SIXINPUTS	int
DELIMITER	String
STARTUP	String
INVALID_INPUT	String
INVALID_DATE	String
INVALID_NAME	String
INVALID_DEP	String
INVALID_SALARY	String
INVALID_PAYRATE	String
INVALID_MGMT	String
INVALID_HOURS_A	String
INVALID_HOURS_B	String
CALCULATED	String
ADDED	String
DUPLICATE	String
SETHOURS	String
REMOVED	String
NONEXISTENT	String
SHUTDOWN	String
ISEMPTY	String
QUIT	String
ADDPARTTIME	String
ADDFULLTIME	String
ADDFULLROLE	String
REMOVE	String
CALCULATE	String
SET	String
PRINTALL	String
PRINTHIRED	String
PRINTDEPART	String
PRINT_HEADER	String
PRINTDEP_HEADER	String
PRINTDATE_HEADER	String
ROUNDING	double
CS	String
ECE	String
IT	String
Consts()	

Date	
month	int
day	int
year	int
Date(String)	
Date()	
isValid()	boolean
isInFuture(int, int, int)	boolean
isLeap()	boolean
isDayValid(boolean)	boolean
compareTo(Date)	int
toString()	String

Profile	
name	String
department	String
dateHired	Date
Profile()	
Profile(String, String, Date)	
isValid()	boolean
getDepartment()	String
getDateHired()	Date
toString()	String
equals(Object)	boolean

Employee	
profile	Profile
periodEarnings	double
Employee()	
Employee(Profile, double)	
getProfile()	Profile
setProfile(Profile)	void
getPeriodEarnings()	double
setPeriodEarnings(double)	void
calculatePayment()	void
toString()	String
equals(Object)	boolean

Fulltime	
annualSalary	double
Fulltime()	
Fulltime(Profile, double)	
getAnnualSalary()	double
setAnnualSalary(double)	void
getPeriodEarnings()	double
setPeriodEarnings(double)	void
toString()	String
equals(Object)	boolean
calculatePayment()	void

Parttime	
hourlyPay	double
hoursWorked	int
Parttime()	
Parttime(Profile, double, int)	
getPeriodEarnings()	double
setPeriodEarnings(double)	void
setHoursWorked(int)	void
getHoursWorked()	int
toString()	String
equals(Object)	boolean
calculatePayment()	void

Company	
emplist	Employee[]
numEmployee	int
Company()	
isEmpty()	boolean
find(Employee)	int
grow()	void
add(Employee)	boolean
remove(Employee)	boolean
setHours(Employee)	boolean
processPayments()	void
print()	void
printByDepartment()	void
sortByDepartment(Employee[])	void
printByDate()	void
sortByDate(Employee[])	void

Management	
bonusPay	double
code	int
Management()	
Management(Profile, double, int)	
getAnnualSalary()	double
setAnnualSalary(double)	void
getPeriodEarnings()	double
setPeriodEarnings(double)	void
toString()	String
equals(Object)	boolean
calculatePayment()	void

PayrollProcessing	
run()	void
addPartTime(String[], Company)	void
addFullTime(String[], Company)	void
addFullRole(String[], Company)	void
removeEmployee(String[], Company)	void
calculate(String[], Company)	void
setHours(String[], Company)	void
validateSharedInput(String, String, Date)	void
validateSalary(double)	void
validatePayRate(double)	void
validateCode(int)	void
validateHours(int)	void
printout(String)	void

CompanyTest	
company	Company
add()	void
remove()	void
setHours()	void

RunProject2	
main(String[])	void