Serializable (Interface)

Comparable (Interface)
- m compareTo(T) : int

Enum (Class)
- f name : String
- f ordinal : int
- m Enum(String, int)
- m name() : String
- m ordinal() : int
- m toString() : String
- m equals(Object) : boolean
- m hashCode() : int
- m clone() : Object
- m compareTo(E) : int
- m getDeclaringClass() : Class<E>
- m valueOf(Class<T>, String) : T
- m finalize() : void
- m readObject(ObjectInputStream) : void
- m readObjectNoData() : void

EmployeeType (Enum)
- OPERADOR
- SUPERVISOR
- DIRECTOR

EmployeeStatus (Enum)
- AVAILABLE
- BUSY

FunctionalInterface (Annotation)

Runnable (Interface)
- m run() : void

CallService (Interface)
- m run() : void

CallServiceImpl (Class)
- f logger : Logger
- f MIN_CALL_TIME : Integer
- f MAX_CALL_TIME : Integer
- f id : Long
- f employee : Employee
- f customer : Customer
- m CallServiceImpl(Long, Customer, Employee)
- m getRandomTime(Integer, Integer) : Long
- m run() : void

IncomingCall (Interface)
- m addEmployeeToShift(Employee) : void
- m findAllEmployeesInShift() : Map<String, Deque<Employee>>
- m findAllCustomersQueued() : Deque<Customer>
- m addCustomer(Customer) : void

IncomingCallImpl (Class)
- f employeesInShift : Map<String, Deque<Employee>>
- f queuedCustomers : Deque<Customer>
- m IncomingCallImpl()
- m addEmployeeToShift(Employee) : void
- m findAllEmployeesInShift() : Map<String, Deque<Employee>>
- m findAllCustomersQueued() : Deque<Customer>
- m addCustomer(Customer) : void

DispatcherService (Interface)
- m dispatchCall(Long, Customer) : void
- m finishDispatch() : void

DispatcherServiceImpl (Class)
- f logger : Logger
- f MAX_NUMBER_SIM_CALLS : int
- f executor : ExecutorService
- f employeeService : EmployeeService
- f customerService : CustomerService
- m DispatcherServiceImpl(EmployeeService, CustomerService)
- m dispatchCall(Long, Customer) : void
- m finishDispatch() : void

Person (Class)
- f name : String
- m getName() : String

Employee (Class)
- f employeeStatus : EmployeeStatus
- f employeeType : EmployeeType
- m Employee(String, EmployeeType)
- m getEmployeeStatus() : EmployeeStatus
- m setEmployeeStatus(EmployeeStatus) : void
- m getEmployeeType() : EmployeeType
- m toString() : String

Customer (Class)
- m Customer(String)
- m toString() : String

EmployeeService (Interface)
- m findAvailableEmployee() : Employee
- m findEmployeesByStatus(EmployeeStatus) : List<Employee>

EmployeeServiceImpl (Class)
- f logger : Logger
- f shiftEmployees : Map<String, Deque<Employee>>
- m EmployeeServiceImpl(IncomingCall)
- m findAvailableEmployee() : Employee
- m findEmployeesByStatus(EmployeeStatus) : List<Employee>

CustomerService (Interface)
- m getCustomer() : Customer
- m addCustomer(Customer) : void

CustomerServiceImpl (Class)
- f customers : Deque<Customer>
- m CustomerServiceImpl(IncomingCall)
- m getCustomer() : Customer
- m addCustomer(Customer) : void