

# Employees and Logistics Menu Instructions

## Commands

The following commands are supported:

- back.
- create employee.
- get employee.
- delete employee.
- list employees.
- add shift preference.
- delete shift preference.
- create shift.
- add shift staff.
- delete shift staff.
- update shift required role.
- list shifts.
- can work.
- create delivery.
- add delivery source.
- add delivery destination.
- add delivery product.
- delete delivery product.
- delete delivery.
- list deliveries.
- create vehicle.
- list vehicles.
- add driver license.
- update delivery weight.

All example commands should work when using the provided sample data.

## Back

Go back to the main menu.

Any user, including unauthenticated users, can perform this command.

## Usage

quit

## Example

quit

## Create Employee

Add a new employee.

Any user, including unauthenticated users, can perform this command.

### Usage

```
create employee <name> <id> <bank> <bank-id> <bank-branch> <salary-per-hour> <role>
```

Where role is one of the following:

- Logistics.
- HumanResources.
- Stocker.
- Cashier.
- LogisticsManager.
- ShiftManager.
- Driver.
- StoreManager.

### Example

```
create employee Noga 110 Bar 12345 1 30 HumanResources
```

## Get Employee

Get the details of an employee.

Authenticated employees can look themselves up, employees authenticated as HumanResources role can look up all employees.

### Usage

```
get employee <id>
```

### Example

```
get employee 999871163
```

## Delete Employee

Delete an employee.

Only employees authenticated as HumanResources role may invoke this command.

### Usage

```
delete employee <id>
```

### **Example**

```
delete employee 614
```

### **List Employees**

List employees and their roles.

Lists in CSV format for easy import to Excel, to allow incremental rollout of the new system.

Only employees authenticated as HumanResources or StoreManager role may invoke this command.

### **Usage**

```
list employees
```

### **Example**

```
list employees
```

### **Add Shift Preference**

Add an available shift for an employee.

Employees can only invoke this command for themselves.

### **Usage**

```
add shift preference <id> <shift>
```

Where `shift` is one of the following:

- SundayMorning.
- SundayEvening.
- MondayMorning.
- MondayEvening.
- TuesdayMorning.
- TuesdayEvening.
- WednesdayMorning.
- WednesdayEvening.
- ThursdayMorning.
- ThursdayEvening.
- FridayMorning.
- FridayEvening.
- SaturdayMorning.
- SaturdayEvening.

### Example

```
add shift preference 999871163 FridayEvening
```

### Delete Shift Preference

Remove an available shift for an employee.

Employees can only invoke this command for themselves.

### Usage

```
delete shift preference <id> <shift>
```

Where `shift` is one of the following:

- SundayMorning.
- SundayEvening.
- MondayMorning.
- MondayEvening.
- TuesdayMorning.
- TuesdayEvening.
- WednesdayMorning.
- WednesdayEvening.
- ThursdayMorning.
- ThursdayEvening.
- FridayMorning.
- FridayEvening.
- SaturdayMorning.
- SaturdayEvening.

### Example

```
delete shift preference 999871163 FridayEvening
```

### Create Shift

Create a shift.

It is recommended to use the `can work` or `can work with role` commands to pick employees for a shift before creating it.

Only employees authenticated as HumanResources role may invoke this command.

### Usage

```
create shift <date> <type> [staff]
```

Where `type` is one of the following:

- Morning.
- Evening.

### Example

```
create shift 2022-04-24 Evening 999368814 999849854 999481773 999205214 999072804
```

### Add Shift Staff

Add an employee to a shift's staff.

Only employees authenticated as HumanResources role may invoke this command.

### Usage

```
add shift staff <date> <type> <employee-id>
```

Where `type` is one of the following:

- Morning.
- Evening.

### Example

```
add shift staff 2022-04-24 Evening 999871163
```

### Delete Shift Staff

Remove an employee from a shift's staff.

Only employees authenticated as HumanResources role may invoke this command.

### Usage

```
delete shift staff <date> <type> <employee-id>
```

Where `type` is one of the following:

- Morning.
- Evening.

### Example

```
delete shift staff 2022-04-24 Evening 999871163
```

### **Update Required Role In Shift**

Update a shift's required roles.

Only employees authenticated as HumanResources role may invoke this command.

#### **Usage**

```
update shift required role <date> <type> <role> <required-count>
```

Where type is one of the following:

- Morning.
- Evening.

#### **Example**

```
update shift required role 2022-04-24 Evening Stocker 2
```

### **List Shifts**

List all past and future shifts.

Lists shifts in YAML format, for easy parsing in printing scripts. Allowing the customer to incrementally roll out the new system by partially working with paper forms.

Only employees authenticated as HumanResources or StoreManager role may invoke this command.

#### **Usage**

```
list shifts
```

#### **Example**

```
list shifts
```

### **Can Work**

List employees who can work a specified shift.

Lists in CSV format for easy import to Excel, to allow incremental rollout of the new system.

Only employees authenticated as HumanResources or StoreManager role may invoke this command.

## Usage

`can work <shift>`

Where `shift` is one of the following:

- `SundayMorning.`
- `SundayEvening.`
- `MondayMorning.`
- `MondayEvening.`
- `TuesdayMorning.`
- `TuesdayEvening.`
- `WednesdayMorning.`
- `WednesdayEvening.`
- `ThursdayMorning.`
- `ThursdayEvening.`
- `FridayMorning.`
- `FridayEvening.`
- `SaturdayMorning.`
- `SaturdayEvening.`

## Example

`can work MondayEvening`

## Can Work With Role

List employees who can work at a specified shift and are of a specified role.

Lists in CSV format for easy import to Excel, to allow incremental rollout of the new system.

Only employees authenticated as `HumanResources` or `StoreManager` role may invoke this command.

## Usage

`can work with role <shift> <role>`

Where `shift` is one of the following:

- `SundayMorning.`
- `SundayEvening.`
- `MondayMorning.`
- `MondayEvening.`
- `TuesdayMorning.`
- `TuesdayEvening.`
- `WednesdayMorning.`
- `WednesdayEvening.`
- `ThursdayMorning.`

- ThursdayEvening.
- FridayMorning.
- FridayEvening.
- SaturdayMorning.
- SaturdayEvening.

And where role is one of the following:

- Logistics.
- HumanResources.
- Stocker.
- Cashier.
- LogisticsManager.
- ShiftManager.
- Driver.
- StoreManager.

### Example

can work with role MondayEvening HumanResources

### Create Delivery

Create a new delivery.

### Usage

create delivery <registration> <date> <time> <driver-id> <duration-in-minutes>

### Add Delivery Source

Add source site to delivery.

### Usage

add delivery source <id> <contact-name> <contact-phone> <area> <city> <street> <house-number>

### Add Delivery Destination

Add destination site to delivery.

### Usage

delete delivery source <id> <contact-name> <contact-phone> <area> <city> <street> <house-number>

### Add Delivery Product

Add product to delivery.



### Usage

add delivery product <id> <product-id> <count>

### Delete Delivery Product

Remove product from delivery.

### Usage

### List Deliveries

Print a list of all deliveries.

### Usage

delete delivery product <id> <product-id> <count>

### Create Vehicle

Add a delivery vehicle.

### List Vehicles

Print a list of all delivery vehicles.

### Usage

create vehicle <license> <registration> <model> <weight> <max-weight>

### Add Driver License

Add driver license to currently logged in user.

### Usage

add driver license <license>

### Update Delivery Weight

Update weight of delivery before departure.

### Usage

update delivery weight <id> <weight>