

FACULTAD DE INGENIERÍA

GRUPO 5

José Rojas

Tomás Shilton

Exercise 2 ADT

**Index**

**Camera 2**

**Accessory 2**

**Manufacturer 3**

**LineSale 3**

**Sale 4**

**RegisteredUser 5**

**Provider 6**

**ADT: Camera**

**Description**: an object used to take photographs with certain technical specifications which can be sold as a product.

Constructor

**Camera**: name X code X megaPixels X screenSize X opticalZoom X type → Camera

**Comment**: creates a camera with the specified information and technical specifications.

**Precondition**: a name, code (10 characters), type (reflex or compact) and real numbers>0 for the price, mega pixels, screen size, optical zoom.

**Postcondition**: the camera with the specified information.

Modifier

**modifyPrice**: Camera X price → Camera

**Comment**: changes the camera’s price.

**Precondition**: a real number > 0.

**Postcondition**: the camera with the new price.

**ADT: Accessory**

**Description**: a product sold apart from the camera for it.

Constructor

**Accessory**: name X code X accessoryType X comment → Accessory

**Comment**: creates an accessory with the specified information.

**Precondition**: a name, code (10 characters), accessory type, comment and a real number > 0 for the price.

**Postcondition**: the accessory with the specified information.

Modifier

**modifyPrice**: Accessory X price → Accessory

**Comment**: changes the accessory’s price.

**Precondition**: a real number > 0.

**Postcondition**: the accessory with the new price.

**ADT: Manufacturer**

**Description**: makes cameras and accessories.

Constructor

**Manufacturer**: name X code → Manufacturer

**Comment**: creates a manufacturer with the specified information.

**Precondition**: a name and a code (10 characters).

**Postcondition**: the manufacturer with the specified information.

Analyzer

**makeCamera**: name X code X megaPixels X screenSize X opticalZoom X type → Camera

**Comment**: makes a camera with the technical specifications.

**Precondition**: a name, code (10 characters), type and a real number > 0 for the megapixels, screen size and optical zoom.

**Postcondition**: a camera with the specified technical specifications.

**makeAccessory**: name X code X accessoryType X comment → Accessory

**Comment**: makes an accessory with the technical specifications.

**Precondition**: a name, code (10 characters), type and a comment.

**Postcondition**: an accessory with the specified information.

**ADT: LineSale**

**Description**: buying order for a camera or accessory which contains the number of it to be bought.

Constructor

**LineSale**: code X Camera X quantity → LineSale

**Comment**: creates a line sale for a camera with the specified information.

**Precondition**: a code (10 characters), a camera and an integer > 0 for the quantity of cameras.

**Postcondition**: the line sale with the specified information.

**LineSale**: code X Accessory X quantity → LineSale

**Comment**: creates a line sale for an accessory with the specified information.

**Precondition**: a code (10 characters), an accessory and an integer > 0 for the quantity of accessories.

**Postcondition**: the line sale with the specified information.

Modifier

**addCameras**: LineSale X quantity → LineSale

**Comment**: adds the specified number of cameras to the line sale.

**Precondition**: an integer > 0.

**Postcondition**: the line sale with the new number of cameras.

**removeCameras**: LineSale X quantity → LineSale

**Comment**: removes the specified number of cameras from the line sale.

**Precondition**: an integer > 0.

**Postcondition**: the line sale with the new number of cameras.

**addAccessories**: LineSale X quantity → LineSale

**Comment**: adds the specified number of accesories to the line sale.

**Precondition**: an integer > 0.

**Postcondition**: the line sale with the new number of accessories.

**removeAccessories**: LineSale X quantity → LineSale

**Comment**: removes the specified number of accessories from the line sale.

**Precondition**: an integer > 0.

**Postcondition**: the line sale with the new number of accessories.

Analyzer

**getPrice**: LineSale → price

**Comment**: gets the line sale price by multiplying the camera’s or accessory’s price by the quantity of them.

**Precondition**: a line sale.

**Postcondition**: a real number > 0.

**ADT: Sale**

**Description**: a camera line sale and accessory line sale holder, where a customer can check the products he is going to buy and how many of them.

Constructor

**Sale**: id → Sale

**Comment**: creates an empty sale with a total price of 0.

**Precondition**: an id.

**Postcondition**: a sale with the specified id and a total price of 0.

Modifier

**addCameras**: Sale X CameraLineSale → Sale

**Comment**: adds a camera line sale and recalculates the total value of the sale by adding the value of the line sale to it.

**Precondition**: a camera line sale.

**Postcondition**: the sale with the camera line sale added and its price added to the sale’s total price.

**removeCameras**: Sale X Camera X quantityToRemove → Sale

**Comment**: looks for the line sale of the specified camera and removes the quantity specified. Recalculates the total value of the sale with the modified camera line sale. If the line sale product quantity is 0, removes it from the sale.

**Precondition**: a camera and an integer > 0.

**Postcondition**: the sale with the cameras removed and its total price reduced.

**addAccessories**: Sale X AccessoryLineSale → Sale

**Comment**: adds an accessory line sale and recalculates the total value of the sale by adding the value of the line sale to it.

**Precondition**: an accessory line sale.

**Postcondition**: the sale with the accessory line sale added and its price added to the sale’s total price.

**removeAccessories**: Sale X Accessory X quantityToRemove → Sale

**Comment**: looks for the line sale of the specified accessory and removes the quantity specified. Recalculates the total value of the sale with the modified acccessory line sale. If the line sale product quantity is 0, removes it from the sale.

**Precondition**: an accessory and an integer > 0.

**Postcondition**: the sale with the accessories removed and its total price reduced.

**ADT: RegisteredUser**

**Description**: a person who can buy cameras and accessories.

Constructor

**RegisteredUser**: name X id X address X phoneNumber X locality X province X country zipCode X role → RefisteredUser

**Comment**: creates a registered user with the specified information.

**Precondition**: a name, id (9 characters), address, phone number (9 characters), locality, province, country, zip code (5 characters) and a role.

**Postcondition**: the registered user with the specified information.

Modifier

**buyProducts**: RegisteredUser X Sale → RegisteredUser

**Comment**: buys products through a sale.

**Precondition**: a sale.

**Postcondition**: the registered user.

**ADT: Provider**

**Description**: provides cameras and accesories.

Constructor

**Provider**: cif X name X phoneNumber X faxNumber X responsible X address X locality X province X country X zipCode → Provider

**Comment**: creates a provider with the specified information with no cameras and accessories.

**Precondition**: a cif (9 characters), name, phone number (9 characters), fax number (9 characters), responsible, address, locality, province, country and zip code (5 characters).

**Postcondition**: the provider with the specified information and no cameras and accessories.

Modifier

**addCameras**: Provider X Cameras → Provider

**Comment**: adds cameras to the provider’s warehouse.

**Precondition**: a list of cameras.

**Postcondition**: the provider with the cameras added.

**removeCameras**: Provider X Camera → Provider

**Comment**: checks if the camera is in the warehouse and if it is, it is removed.

**Precondition**: a camera.

**Postcondition**: if the camera is in the warehouse, the provider without it. Else, the provider as before the method was applied.

**provideCameras**: Provider X cameraName X quantity → Cameras

**Comment**: receives a camera name and the wanted number of them and, if available, returns them.

**Precondition**: a name and an integer > 0.

**Postcondition**: if enough products which satisfy the requirements are found, a list of those cameras. Else, nothing.

**addAccessories**: Provider X Accessories → Provider

**Comment**: adds accessories to the provider’s warehouse.

**Precondition**: a list of accessories.

**Postcondition**: the provider with the accessories added.

**removeAccessories**: Provider X Accessory → Provider

**Comment**: checks if the accessory is in the warehouse and if it is, it is removed.

**Precondition**: an accessory.

**Postcondition**: if the accessory is in the warehouse, the provider without it. Else, the provider as before the method was applied.

**provideAccessories**: Provider X Accessory X quantity → Accessories

**Comment**: receives an accessory and the wanted number of them and, if available, returns them.

**Precondition**: an accessory and an integer > 0.

**Postcondition**: if enough products which satisfy the requirements are found, a list of those accessories. Else, nothing.

Analyzer

**getCameraQuantity**: Provider X cameraName → numberOfCameras

**Comment**: given a camera name, returns the quantity of that camera in the warehouse.

**Precondition**: a camera name.

**Postcondition**: an integer >= 0.

**getAccessoryQuantity**: Provider X Accessory → numberOfAccessories

**Comment**: given an accessory, returns the quantity of it in the warehouse.

**Precondition**: an accessory.

**Postcondition**: an integer >= 0.