

<<Enumeration>> SortOrder
ASC
DESC

<<Enumeration>> SortField
BRAND
MODEL
DISPLAY_SIZE_INCHES
PRICE

<<Enumeration>> OrderStatus
NEW
IN_DELIVERY
REJECTED
DELIVERED

<<Interface>> PhoneService
+getPhoneByKey(key: Long) : Optional<Phone> +getPhoneByModel(model: String) : Optional<Phone> +savePhone(phone: Phone): void +getPhones(offset: int, limit: int): List<Phone> +getPhones(paramsForSearch: ParamsForSearch): List<Phone> +countPhoneQuantity(paramsForSearch: ParamsForSearch): Long +getHotPricesPhones(amount: int): List<Phone>

ProductListPageController
-QUANTITY_ON_PAGE: Long -phoneServiceImpl: PhoneService -cartService: CartService -httpSession: HttpSession +showProductList(search: String, field: String, order: String, page: Long, model: Model): String

OrderPageController
-orderService: OrderService -environment: Environment -cartService: CartService -httpSession: HttpSession -orderDataDtoValidator: OrderDataDtoValidator +getOrder(model: Model): String +placeOrder(orderDataDto: OrderDataDto, model: Model, bindingResult: BindingResult, redirectAttributes: String) -prepareModelForValidationErrors(bindingResult: BindingResult, redirectAttributes: RedirectAttributes): String -prepareModelForEmptyCart(redirectAttributes: RedirectAttributes): String -showCartAsOrder(cart: Cart, model: Model): void

CartPageController
-REDIRECT_TO_CART_PAGE: String -phoneServiceImpl: PhoneService -cartService: CartService -httpSession: HttpSession -phoneArrayDtoValidator: PhoneArrayDtoValidator +getCart(successDelete: boolean, successUpdate: boolean, isOutOfStock: boolean, error: boolean, errorsId: List<Long>, model: Model): String +updateCart(phoneArrayDto: PhoneArrayDto, model: Model, bindingResult: BindingResult): String +deleteFromCart(id: Long, model: Model): String -prepareModelForEmptyCart(cart: Cart, model: Model): String -validationFailed(cart: Cart, bindingResult: BindingResult, model: Model): String

CartItem
-phone: Phone -quantity: Long -price: BigDecimal +CartItem(phone: Phone, quantity: Long, price: BigDecimal) +setPhone(phone: Phone): void +getPhone(): Phone +setQuantity(quantity: Long): void +getQuantity(): Long +getPrice(): BigDecimal +setPrice(price: BigDecimal)

<<Interface>> CartService
+getCart(httpSession: HttpSession) : Cart +addPhone(phoneId: Long, quantity: Long, cart: Cart): void +update(items: Map<Long, Long>, cart: Cart): void +remove(phoneId: Long, cart: Cart): void +deleteCart(httpSession: HttpSession): void

<<Interface>> OrderService
+placeOrder(cart: Cart, orderDataDto: OrderDataDto, deliveryPrice: Long) : Long +getOrders(limit: int, offset: int) : List<Order> +updateStatus(orderStatus: OrderStatus, orderId: Long): void +getOrderById(id: Long): Optional<Order>

Cart
-cartItems: List<CartItem> -totalQuantity: Long -totalCost: BigDecimal +Cart() +setCartItems(cartItems: List<CartItem>): void +getCartItems(): List<CartItem> +setTotalQuantity(totalQuantity: Long): void +getTotalQuantity(): Long +setTotalCost(totalCost: BigDecimal): void +getTotalCost(): BigDecimal

PhoneServiceImpl
-jdbcPhoneDao: PhoneDao +getPhoneByKey(key: Long) : Optional<Phone> +getPhoneByModel(model: String) : Optional<Phone> +savePhone(phone: Phone): void +getPhones(offset: int, limit: int): List<Phone> +getPhones(paramsForSearch: ParamsForSearch): List<Phone> +countPhoneQuantity(paramsForSearch: ParamsForSearch): Long +getHotPricesPhones(amount: int): List<Phone>

OrderServiceImpl
-jdbcOrderDao: OrderDao -jdbcStockDao: StockDao +placeOrder(cart: Cart, orderDataDto: OrderDataDto, deliveryPrice: Long) : Long +getOrders(limit: int, offset: int) : List<Order> +updateStatus(orderStatus: OrderStatus, orderId: Long): void +getOrderById(id: Long): Optional<Order>

HttpSessionCartService
-jdbcPhoneDao: OrderDao -jdbcStockDao: StockDao +getCart(httpSession: HttpSession) : Cart +addPhone(phoneId: Long, quantity: Long, cart: Cart): void +update(items: Map<Long, Long>, cart: Cart): void +remove(phoneId: Long, cart: Cart): void +deleteCart(httpSession: HttpSession): void +addToCart(quantity: Long, stock: Stock, phone: Phone, cart: Cart): void +checkQuantity(phoneId: Long, quantityDifference: Long): boolean +findCartItem(phoneId: Long, cart: Cart): Optional<CartItem> +calculateCart(cart: Cart): void

OrderDataDto
-firstName: String -lastName: String -address: String -phone: String -additionalInfo: String +setFirstName(firstName: String): void +getFirstName(): String +setLastName(lastName: String): void +getLastName(): String +setAddress(address: String): void +getAddress(): String +setPhone(phone: String): void +getPhone(): String +setAdditionalInfo(additionalInfo: String): void +getAdditionalInfo(): String

<<Interface>> PhoneDao
+get(key: Long) : Optional<Phone> +get(model: String) : Optional<Phone> +save(phone: Phone): void +findAll(offset: int, limit: int): List<Phone> +findAll(paramsForSearch: ParamsForSearch): List<Phone> +count(paramsForSearch: ParamsForSearch): Long +findMaxDiscountPercentPhones(amount: int): List<Phone>

<<Interface>> OrderDao
+get(key: Long) : Optional<Order> +save(phone: Order): Long +getOrders(offset: int, limit: int): List<Order> +updateStatus(orderStatus: orderStatus, key: Long): void

PhoneArrayDto
-phoneId: String[] -quantity: String[] +setPhoneId(phoneId: String[]): void +getPhoneId(): String[] +setQuantity(quantity: String[]): void +setQuantity(): String[]

JdbcPhoneDao
-SQL_INSERT_PHONE : String -SQL_GET_ALL_PHONES : String -SQL_GET_MAX_DISCOUNTED_PHONES : String -SQL_GET_PHONE : String -SQL_SELECT_COUNT_FIND_ALL_EXTENDED: String -SQL_WHERE_SEARCH : String -SQL_SELECT_COLOR_IDS : String -SQL_GET_PHONE_BY_MODEL : String -jdbcTemplate: JdbcTemplate -namedParameterJdbcTemplate: NamedParameterJdbcTemplate -phoneResultSetExtractor: PhoneResultSetExtractor -jdbcColorDao: JdbcColorDao +get(key: Long) : Optional<Phone> +get(model: String) : Optional<Phone> +save(phone: Phone): void +findAll(offset: int, limit: int): List<Phone> +findAll(paramsForSearch: ParamsForSearch): List<Phone> +count(paramsForSearch: ParamsForSearch): Long +findMaxDiscountPercentPhones(amount: int): List<Phone>

JdbcOrderDao
-SELECT_ORDER_SQL : String -INSERT_ORDER_SQL : String -SQL_INSERT_INTO_ORDER_ITEMS : String -SELECT_ORDERS : String -UPDATE_ORDER_STATUS_SQL: String -SQL_NEEDED_ORDER : String -jdbcTemplate: JdbcTemplate -namedParameterJdbcTemplate: NamedParameterJdbcTemplate -orderResultSetExtractor: OrderResultSetExtractor -orderListResultSetExtractor: OrderListResultSetExtractor +get(key: Long) : Optional<Order> +save(phone: Order): Long +getOrders(offset: int, limit: int): List<Order> +updateStatus(orderStatus: orderStatus, key: Long): void

PhoneDto
-phoneId: Long -quantity: Long +setPhoneId(phoneId: Long): void +getPhoneId(): Long +setQuantity(quantity: Long): void +setQuantity(): Long