```
class ItemNotFoundException(Exception):
    pass

class InvalidQuantityException(Exception):
    pass

class ItemAlreadyExistsException(Exception):
    pass

class InventoryManagementSystem:
    def __init__(self):
        self.inventory = {}

    def add_new_item(self, item_name, quantity):
        if item_name in self.inventory:
            raise ItemAlreadyExistsException("Item already exists in the inventory.")
        if quantity < 0:
            raise InvalidQuantityException("Quantity cannot be negative.")
        self.inventory[item_name] = quantity
        print(f"Item {item_name} added with quantity {quantity}.")

    def update_quantity(self, item_name, quantity):
        if item_name not in self.inventory:
            raise ItemNotFoundException("Item not found in the inventory.")
        if quantity < 0:
            raise InvalidQuantityException("Quantity cannot be negative.")
        self.inventory[item_name] += quantity
        print(f"Item {item_name} updated. New quantity: {self.inventory[item_name]}.")

    def generate_report(self):
        if not self.inventory:
```

```
            print("Inventory is empty.")
        else:
            print("Inventory Report:")
            for item_name, quantity in self.inventory.items():
                print(f"Item: {item_name}, Quantity: {quantity}")

# Example usage
def main():
    system = InventoryManagementSystem()

    try:
        system.add_new_item("Apple", 50)
        system.add_new_item("Banana", 30)
    except ItemAlreadyExistsException as e:
        print(e)
    except InvalidQuantityException as e:
        print(e)

    try:
        system.update_quantity("Apple", 20)
        system.update_quantity("Orange", 10)
    except ItemNotFoundException as e:
        print(e)
    except InvalidQuantityException as e:
        print(e)

    system.generate_report()

main()
```