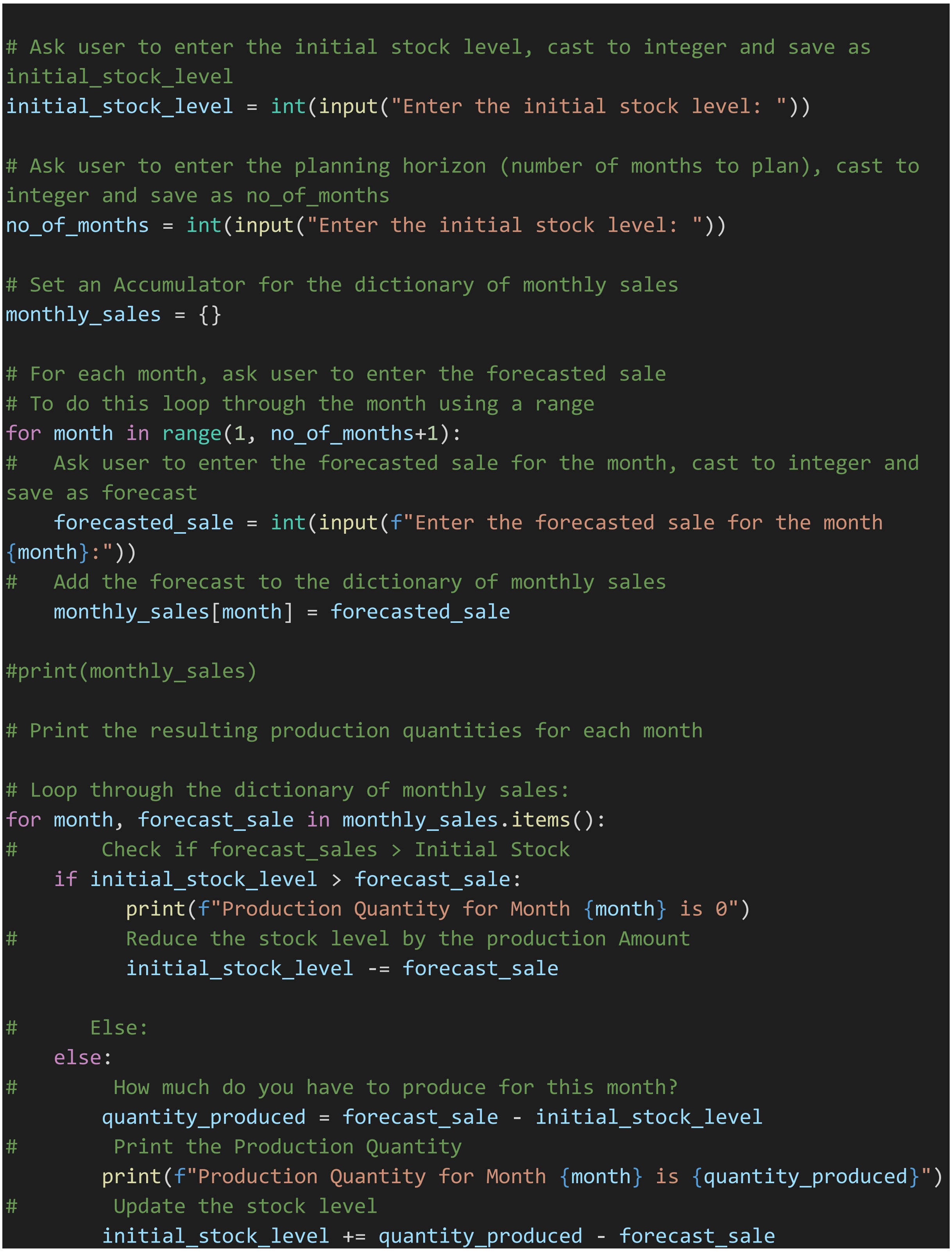

```
# Ask user to enter the initial stock level, cast to integer and save as
initial_stock_level
initial_stock_level = int(input("Enter the initial stock level: "))

# Ask user to enter the planning horizon (number of months to plan), cast to
integer and save as no_of_months
no_of_months = int(input("Enter the initial stock level: "))

# Set an Accumulator for the dictionary of monthly sales
monthly_sales = {}

# For each month, ask user to enter the forecasted sale
# To do this loop through the month using a range
for month in range(1, no_of_months+1):
#   Ask user to enter the forecasted sale for the month, cast to integer and
save as forecast
    forecasted_sale = int(input(f"Enter the forecasted sale for the month
{month}:"))
#   Add the forecast to the dictionary of monthly sales
    monthly_sales[month] = forecasted_sale

#print(monthly_sales)

# Print the resulting production quantities for each month

# Loop through the dictionary of monthly sales:
for month, forecast_sale in monthly_sales.items():
#       Check if forecast_sales > Initial Stock
    if initial_stock_level > forecast_sale:
          print(f"Production Quantity for Month {month} is 0")
#         Reduce the stock level by the production Amount
          initial_stock_level -= forecast_sale

#      Else:
    else:
#        How much do you have to produce for this month?
        quantity_produced = forecast_sale - initial_stock_level
#        Print the Production Quantity
        print(f"Production Quantity for Month {month} is {quantity_produced}")
#        Update the stock level
        initial_stock_level += quantity_produced - forecast_sale
```