CODE

settings.py > app_name

```python
filename="expenses.csv"
budget_file="budget.txt"
app_name="Pocket Guard App"
```

database.py > delete_row

```python
import csv
import os
import settings

def get_all():
    l = []
    if os.path.exists(settings.filename):
        with open(settings.filename, "r") as f:
            r = csv.reader(f)
            for row in r:
                l.append(row)
    return l

def save_one(name, price):
    with open(settings.filename, "a", newline="") as f:
        w = csv.writer(f)
        w.writerow([name, price])

def save_budget_limit(amount):
    with open(settings.budget_file, "w") as f:
        f.write(str(amount))

def get_budget_limit():
    if not os.path.exists(settings.budget_file):
        return 5000.0
    with open(settings.budget_file, "r") as f:
        data = f.read()
        return float(data)
```

```python

def delete_row(line_number):
    rows = get_all()
    idx = line_number - 1

    if idx < 0 or idx >= len(rows):
        return False
    rows.pop(idx)
    with open(settings.filename, "w", newline="") as f:
        w = csv.writer(f)
        w.writerows(rows)
    return True
```

logic.py > calc_total

```python
def calc_total(data):
    t = 0
    for row in data:
        p = float(row[1])
        t = t + p
    return t

def check_warning(total, limit):
    if total > limit:
        return "Warning: Over Budget!!"
    else:
        left = limit - total
        return f"Safe. (Remaining: {left})"
```

interface.py > run_app

```python
import tkinter as tk
from tkinter import simpledialog, messagebox
import database
import logic
import settings

def run_app():

    root = tk.Tk()
    root.title(settings.app_name)
    root.geometry("300x500")

    l1 = tk.Label(root, text="Item Name:")
    l1.pack(pady=5)
    e1 = tk.Entry(root)
    e1.pack()

    l2 = tk.Label(root, text="Price:")
    l2.pack(pady=5)
    e2 = tk.Entry(root)
    e2.pack()

    def btn_add():
        n = e1.get()
        p = e2.get()

        if n == "" or p == "":
            messagebox.showerror("Error", "type something")
            return
        try:
            database.save_one(n, p)
            messagebox.showinfo("Saved", "Added item")
            e1.delete(0, tk.END)
            e2.delete(0, tk.END)
```

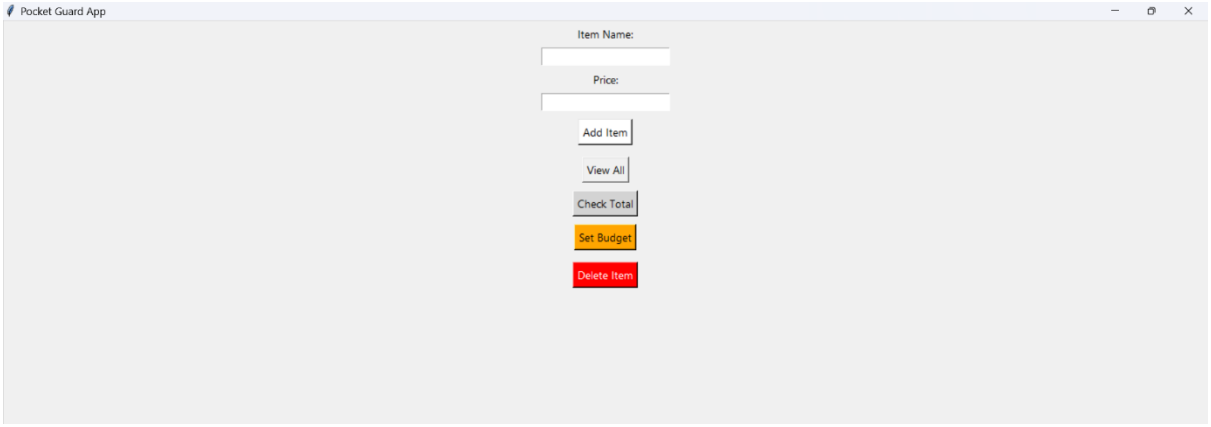
```
            e2.delete(0, tk.END)
        except:
            print("save error")

    def btn_view():
        d = database.get_all()
        text = ""
        i = 1
        for row in d:
            text = text + str(i) + ". " + row[0] + " : " + row[1] + "\n"
            i = i + 1

        messagebox.showinfo("My List", text)

    def btn_total():
        d = database.get_all()
        limit = database.get_budget_limit()
        ans = logic.calc_total(d)
        stat = logic.check_warning(ans, limit)
        msg = f"Total Spent: {ans}\nMy Budget: {limit}\n{stat}"
        messagebox.showinfo("Report", msg)

    def btn_set_budget():
        new_b = simpledialog.askfloat("Budget", "Enter new budget limit:")
        if new_b is not None:
            database.save_budget_limit(new_b)
            messagebox.showinfo("Done", "Budget Updated")

```

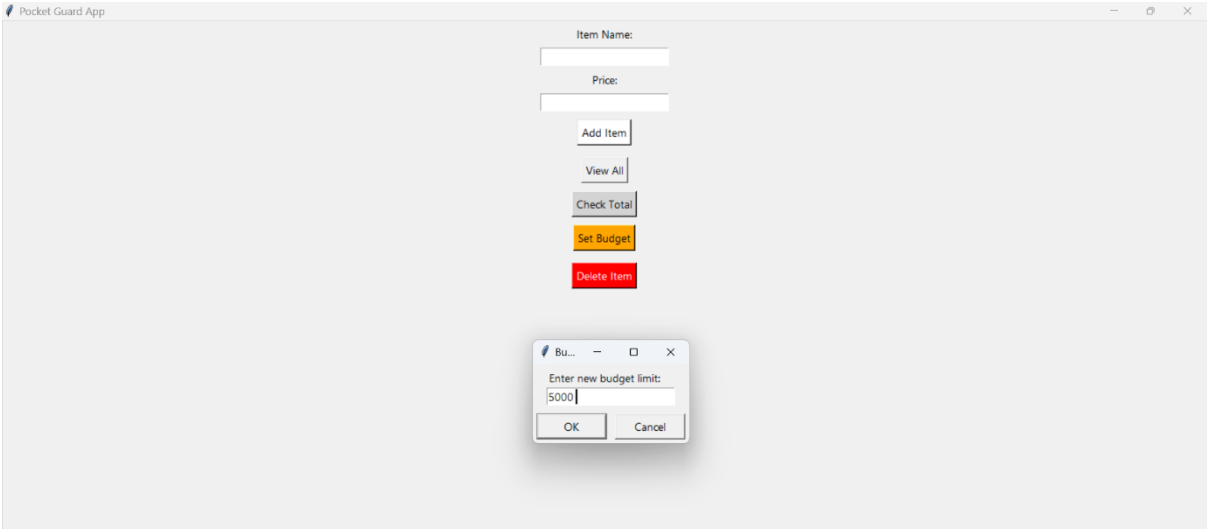
```

    def btn_delete():
        btn_view()
        num = simpledialog.askinteger("Delete", "Enter line number to delete:")

        if num is not None:
            success = database.delete_row(num)
            if success:
                messagebox.showinfo("Deleted", "Item removed")
            else:
                messagebox.showerror("Error", "Invalid line number")

    b1 = tk.Button(root, text="Add Item", command=btn_add, bg="white")
    b1.pack(pady=10)

    b2 = tk.Button(root, text="View All", command=btn_view)
    b2.pack(pady=5)

    b3 = tk.Button(root, text="Check Total", command=btn_total, bg="lightgrey")
    b3.pack(pady=5)

    b4 = tk.Button(root, text="Set Budget", command=btn_set_budget, bg="orange")
    b4.pack(pady=5)

    b5 = tk.Button(root, text="Delete Item", command=btn_delete, bg="red", fg="white")
    b5.pack(pady=10)

    root.mainloop()
```

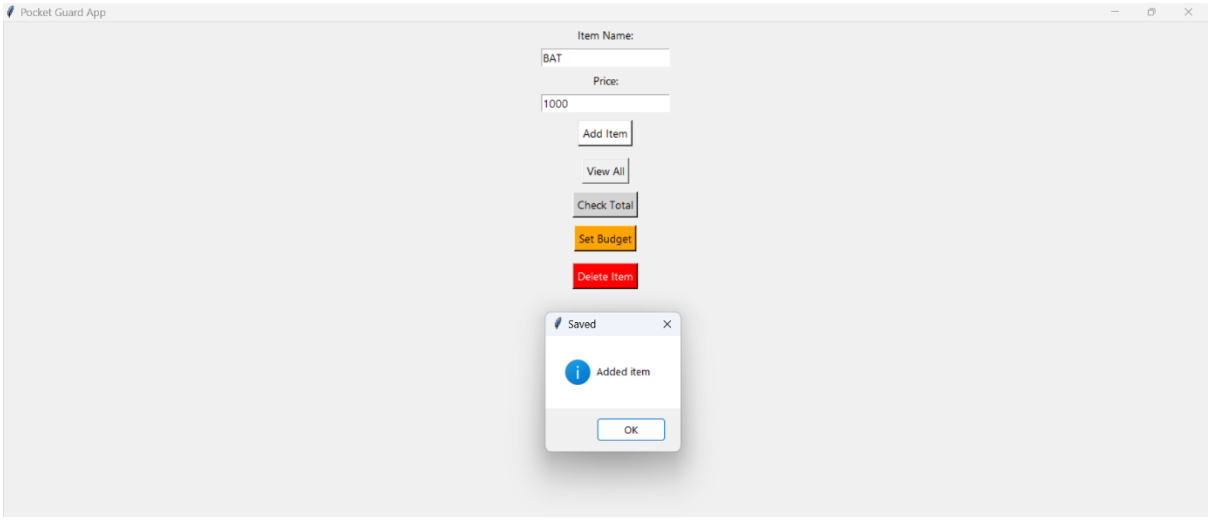
main.py > ...

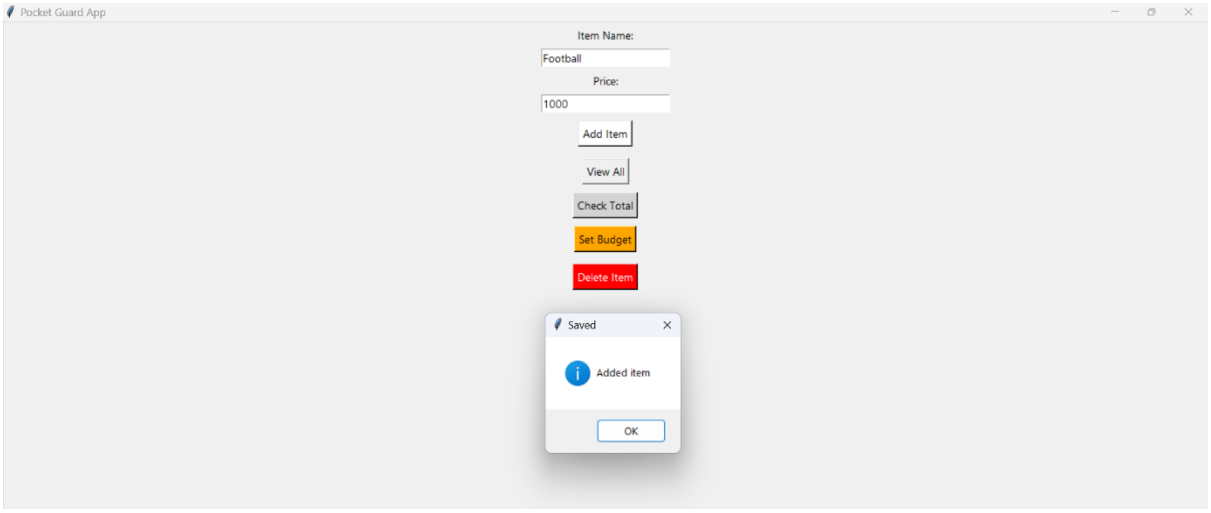
```
import interface
if __name__ == "__main__" :
    interface.run_app()
```

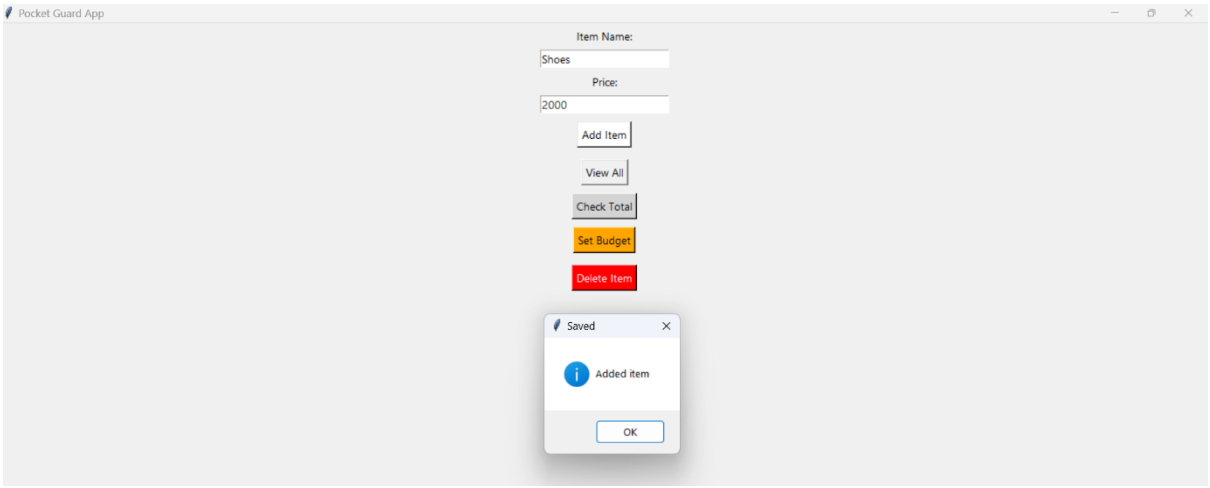
OUTPUT

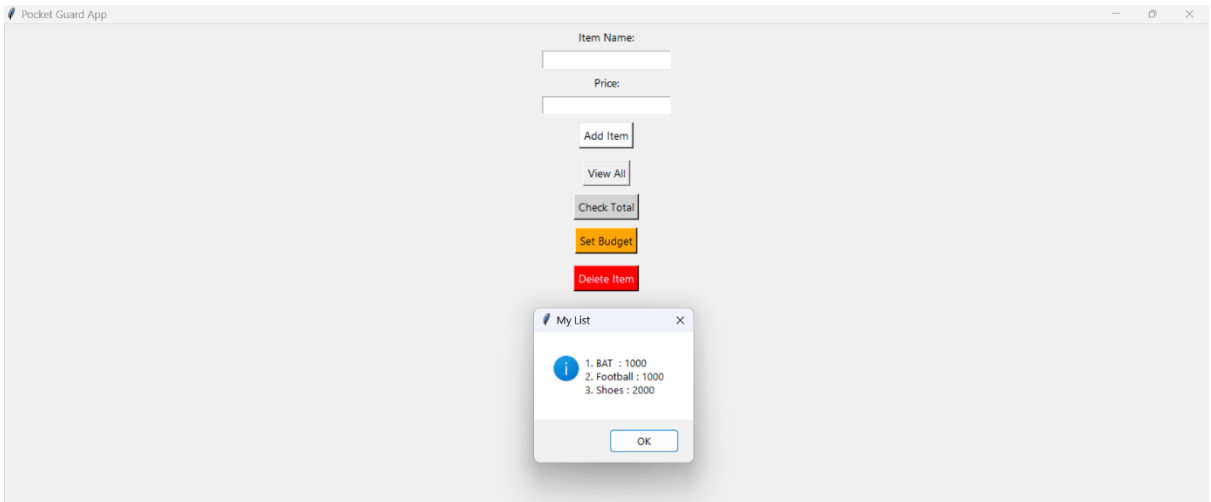

Pocket Guard App
Item Name:
Price:
Add Item
View All
Check Total
Set Budget
Delete Item

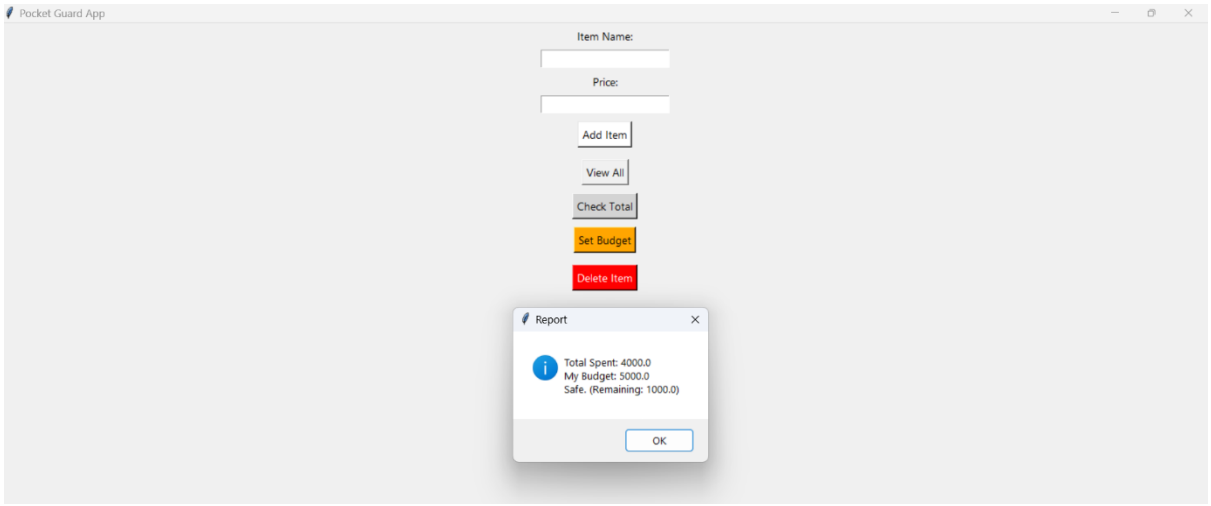

Pocket Guard App
Item Name:
Price:
Add Item
View All
Check Total
Set Budget
Delete Item
Bu...
Enter new budget limit:
5000
OK
Cancel

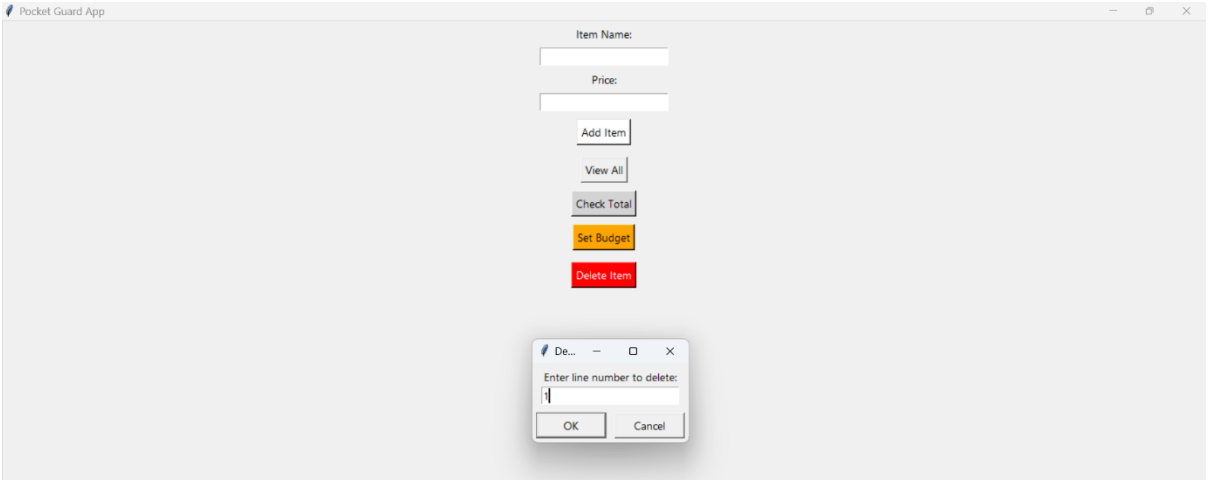

Pocket Guard App
Item Name:
BAT
Price:
1000
Add Item
View All
Check Total
Set Budget
Delete Item
Saved
Added item
OK

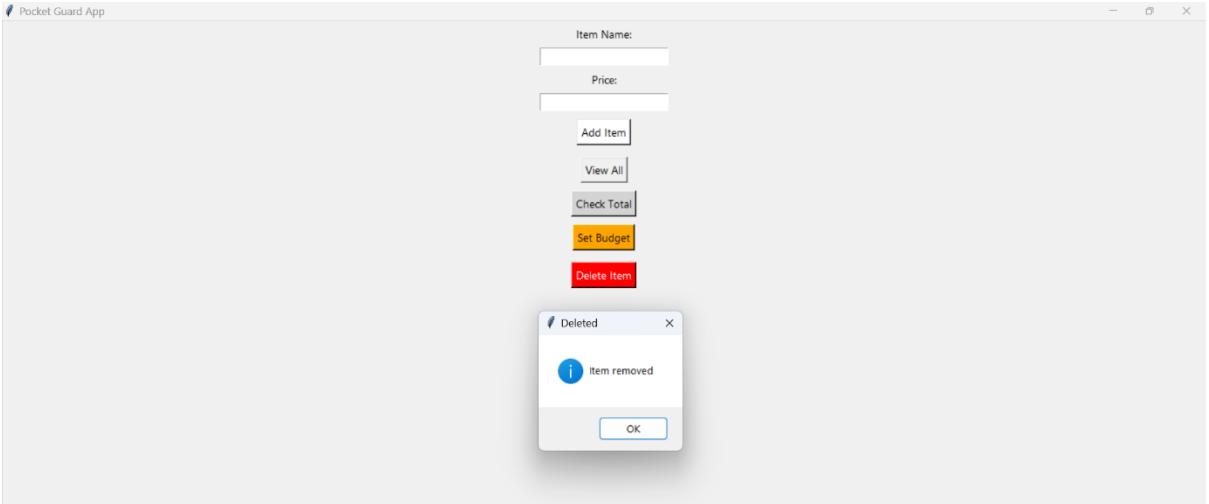

Pocket Guard App
Item Name:
Football
Price:
1000
Add Item
View All
Check Total
Set Budget
Delete Item
Saved
Added item
OK

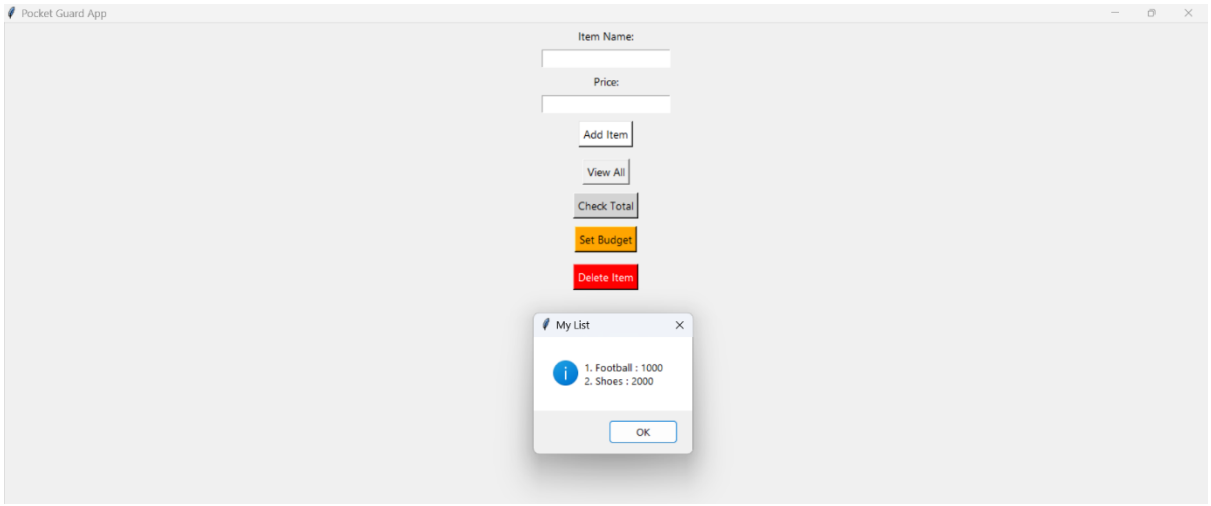

Pocket Guard App
Item Name:
Shoes
Price:
2000
Add Item
View All
Check Total
Set Budget
Delete Item
Saved
Added item
OK


Pocket Guard App
Item Name:
Price:
Add Item
View All
Check Total
Set Budget
Delete Item
My List
1. BAT : 1000
2. Football : 1000
3. Shoes : 2000
OK


Pocket Guard App
Item Name:
Price:
Add Item
View All
Check Total
Set Budget
Delete Item
Report
Total Spent: 4000.0
My Budget: 5000.0
Safe. (Remaining: 1000.0)
OK


Pocket Guard App
Item Name:
Price:
Add Item
View All
Check Total
Set Budget
Delete Item
De...
Enter line number to delete:
1
OK
Cancel


Pocket Guard App
Item Name:
Price:
Add Item
View All
Check Total
Set Budget
Delete Item
Deleted
Item removed
OK


Pocket Guard App
Item Name:
Price:
Add Item
View All
Check Total
Set Budget
Delete Item
My List
1. Football : 1000
2. Shoes : 2000
OK