

Task 1

Разработать класс, определяющий покупку некоторого товара.

Поля:

- название товара;
- цена в рублях с копейками (BYN);
- кол-во единиц товара.

Конструктор:

- с параметрами.

Методы:

- `getCost()` – вычисляет стоимость покупки;
- `toString()` – переводит объект в строку с разделителями «;» (все поля и `getCost()`);

Создать массив (с числовыми индексами) с четырьмя или более покупками. Вывести покупки в браузер через `toString()`.

– В программе реализовать функцию `makeS(покупка)`, которая с использованием замыкания позволяет осуществить вычисление стоимости всех покупок по примерно такому алгоритму:

```
for (i = 0; i < purch.length; i++)  
{  
    makeS(purch[i]); //суммирование  
}  
document.writeln("<h2>Итого : "+makeS()+"</h2>"); //вывод результата
```

После списка покупок вывести итоговую стоимость всех покупок. Желательно стоимость и цену выводить с двумя знаками после запятой (до копеек).

Task 2

Разработать производный от описанного в *Task1* класс, определяющий покупку товара с некоторой скидкой *в процентах*. Использовать **функциональное** наследование. Переопределить методы `getCost()` и `toString()`. В `toString()` порождённого класса вызывать `toString()` базового класса, а не дублировать его функциональность. В массив к имеющимся покупкам базового класса добавить три или больше покупок производного. Код, оставшийся из прошлого задания, должен вывести новый получившейся список покупок и итоговую стоимость внизу.

Task 3

Разработать производный от описанного в *Task1* класс, определяющий покупку товара с некоторой фиксированной скидкой *от общей стоимости*. Использовать наследование **в прототипном** стиле. Переопределить методы `getCost()` и `toString()`. В `toString()` порождённого класса вызывать `toString()` базового класса, а не дублировать его функциональность. В массив к имеющимся покупкам базового класса добавить три или больше покупки производного. Код, оставшийся из прошлого задания, должен вывести новый получившейся список покупок и итоговую стоимость внизу.

Task 4

Объединить разработанные в *Task1–3* классы в одну программу. Должны быть доступны *три* класса:

- базовый класс покупки;
- покупка со скидкой в процентах с функциональным наследованием от базовой покупки;

– покупка со скидкой от общей стоимости с прототипным наследованием от базовой покупки.

1 Создать *исходный* массив из нескольких покупок (9+) различных классов.

2 Вывести исходные покупки, но при этом цикл вывода заменить на метод массива `forEach()`.

3 С использованием метода массива `filter()` отфильтровать покупки, получив новый массив с покупками, в который скопируются только те, у которых итоговая стоимость больше 20. Вывести получившиеся покупки.

4 С помощью метода массива `map()` получить новый массив покупок из исходного, при этом покупки без скидок (базового класса), должны замениться покупками со скидкой в процентах, где процент скидки численно равен удвоенному количеству товара. Остальные покупки должны остаться без изменений. Вывести покупки.

5 С помощью методов массива `every()/some()` проверить, все ли/есть ли покупки с итоговой стоимостью, больше 50. Вывести ответы (да/нет, true/false...)

6 Найти покупку с максимальной стоимостью с помощью метода массива `reduce()`. Вывести её.

7 Получить с помощью метода массива `reduceRight()` новую покупку без скидок со следующими характеристиками:

- название товара – самое длинное из названий всех покупок исходного массива;
- цена – самая высокая цена из всех покупок исходного массива;
- количество – минимальное количество товара во всех покупках исходного массива.

Вывести получившуюся покупку.

Рекомендации по реализации:

Используйте современный браузер, (IE11, Edge, Chrome ...).

Создайте в Visual Studio новый файл (Ctrl+N) «HTML–страница». В части `<body>...</body>` разместите Ваш скрипт. На <https://learn.javascript.ru/> примеры осуществляют вывод через `alert(...)`, однако в заданиях удобнее выводить текст сразу в браузер, через `document.write(...)` / `document.writeln(...)`. Пример:

```
<!DOCTYPE html>
```

```
<html lang="en" xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta charset="utf-8" />
  <title></title>
</head>
<body>
<script>
  document.writeln("<h2>Результаты</h2>");
  document.writeln(10 + 2 * 3);

</script>
</body>
</html>
```

Если неверно отображаются русские буквы, значит, файл **не** сохранён в кодировке *utf-8*. Для исправления воспользуйтесь командой в меню *Файл | Дополнительные параметры сохранения*.