

1. Сегодня мы доделаем то, что не успели в прошлый раз.
2. А не успели мы добавить таймер обратного отсчёта, чтобы игра заканчивалась.



3. Но перед этим нужно, чтобы игра не начиналась неожиданно для игрока. У игрока должна быть возможность запустить её. Для этого мы создадим специальный блок div с приветственным текстом. Остальные блоки мы спрячем, так как они должны появиться только после старта игры. Прятать мы их будем с помощью команды `.style.display = 'none'`

**Создадим объект для приветственного сообщения и спрячем в начале все другие блоки**

```
document.getElementsByTagName('body')[0].insertAdjacentHTML('beforeEnd',
  '<div id="helloText">Нажмите клавишу enter чтобы начать игру</div>');
var helloText = document.getElementById('helloText');
helloText.style.display = 'block';
helloText.style.textAlign = "center";
helloText.style.fontSize = 72+"pt";
```

```
document.getElementsByTagName('body')[0].insertAdjacentHTML('beforeEnd',
  '<div id="score">Очки: 0</div>');
var scoreObj = document.getElementById('score');
scoreObj.style.textAlign = "center";
scoreObj.style.fontSize = 72+"pt";
var score=0;
scoreObj.style.display = 'none';

document.getElementsByTagName('body')[0].insertAdjacentHTML('beforeEnd',
  '<div id="orange"><img alt="orange" data-bbox="168 241 198 266" data-kind="parent" style="vertical-align: middle;"/></div>');
var orange = document.getElementById('orange');
orange.style.position = 'fixed';
orange.style.display = 'none';

document.getElementsByTagName('body')[0].insertAdjacentHTML('beforeEnd',
  '<div id="pig"><img alt="pig" data-bbox="318 234 353 259" data-kind="parent" style="vertical-align: middle;"/></div>');
var pig = document.getElementById('pig');
pig.style.position = 'fixed';
pig.style.display = 'none';
```

4. В самом начале игры у нас не должен работать обработчик передвижения мыши, так что закомментируйте его на время. Добавьте новый обработчик для запуска игры и присоедините его к документу по событию нажатия клавиши.

### Добавим обработчик нажатия клавиши

```
var enterListener = function(event){startGame(event)};
document.addEventListener("keydown",enterListener);

document.addEventListener("mousemove", mouseListener);
```

5. Что же нужно сделать, чтобы игра запустилась? Во-первых, проверить, что была нажата клавиша enter, а не любая другая. Для этого мы сравним код клавиши с 13. 13 – это код клавиши enter.

После этого обнулим счёт, спрячем приветственное окно и покажем все остальные элементы – счёт, апельсин и свинку.

Кроме того, и это важно, мы отключим обработчик нажатия клавиши и включим обработчик перемещения мыши командами `removeEventListener` и `addEventListener` соответственно.

После этого можно смело перемещать апельсинку в случайную точку нашей командой `spawnOrange()`;

### Опишем функцию startGame

```
function startGame(event){
  if (event.keyCode ==13){
    score = 0;
    setScore(0);

    helloText.style.display = 'none';
    scoreObj.style.display = 'block';
    orange.style.display = 'block';
    pig.style.display = 'block';

    document.removeEventListener('keydown', enterListener);
    document.addEventListener("mousemove", mouseListener);

    spawnOrange();
  }
}
```

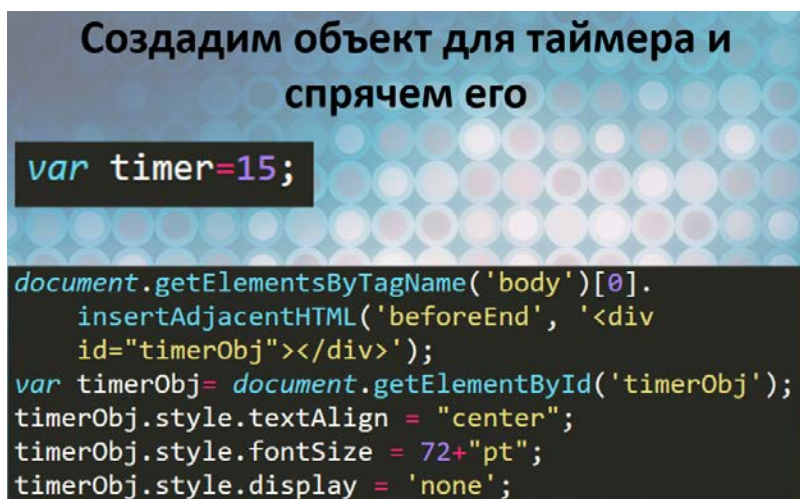
6. На данный момент игра должна начинаться с текста «нажмите клавишу...» и только после нажатия – запускаться. Проверьте, что так и есть.



7. Таймер – это счётчик временных интервалов. Благодаря ему мы можем запускать какие-то действие через определённые промежутки времени. Один из способов реализовать таймер – это воспользоваться командой `setInterval` (команда, время\_в\_мс), которая будет повторять код из первого параметра каждые (количество мс) из второго параметра. Если мы планируем отключать и включать этот таймер (а мы планируем), то лучше завести переменную для хранения ID таймера, чтобы по этому ID его выключать и включать. Для выключения используется команда `clearInterval`. Код на слайде повторять не нужно.



8. Как и раньше, создадим специальный `div` для вывода значения таймера, а также специальную переменную, в которой и будет храниться время, которое осталось игроку. Не забудем, что в начале мы не видим этот блок.



9. Напишем функцию для вывода информации о времени в div блок и запустим её в конце функции StartGame.


**Напишем функцию для отображения таймера и вызовем её в StartGame**

```
function setTimer(timeToSet){
  console.log(timer);
  timerObj.innerHTML = "Время: "+ timeToSet;
}

scoreObj.style.display = 'block';
orange.style.display = 'block';
pig.style.display = 'block';

spawnOrange();
setScore(0);

timerObj.style.display = 'block';
timer = 15;
setTimer(timer);
```



10. Создадим переменную для записи ID таймера, и назначим ей запуск функции cntdwn(); каждую секунду (1000 мс).

**Назначим функцию обратного отсчёта на таймер**

```
var intervalId;

//В startGame
intervalId = setInterval(
  function(){cntdwn();}, 1000
);
```

11. Сама же эта функция (cntdwn) очень проста – уменьшаем оставшееся время на единицу и выводим его в блок div.

**Опишем функцию обратного отсчёта**

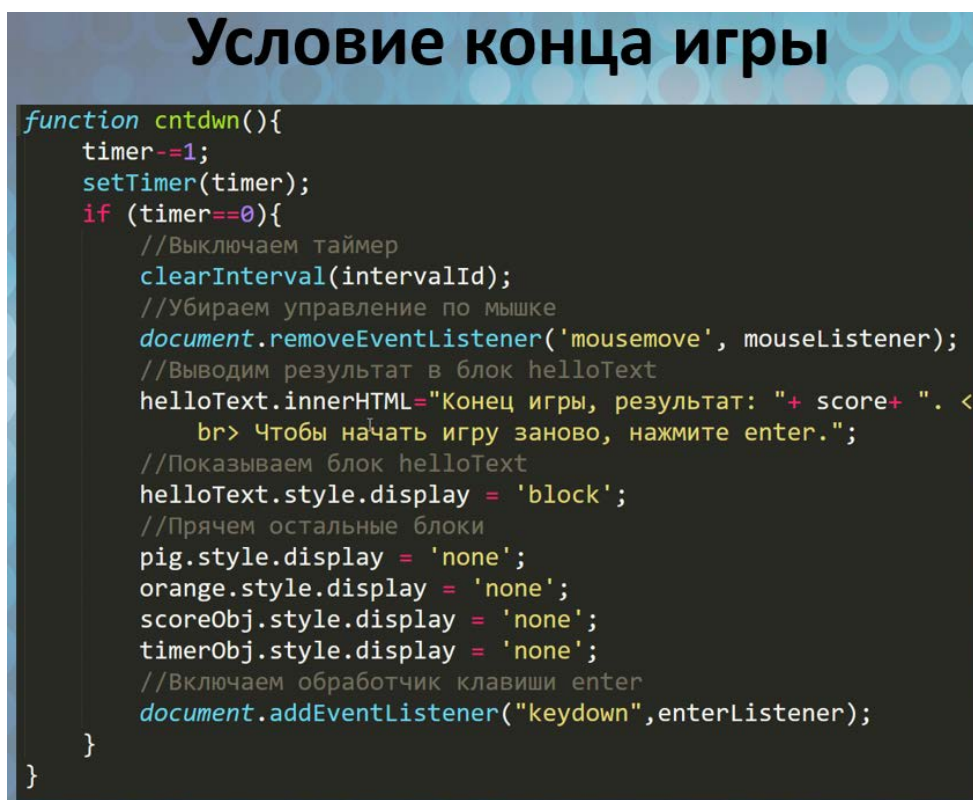
```
function cntdwn(){
  timer-=1;
  setTimer(timer);
}
```

12. На данный момент у нас должно уменьшаться время. Но это уменьшение не прерывается, так как у нас нет проверки на истечение времени.

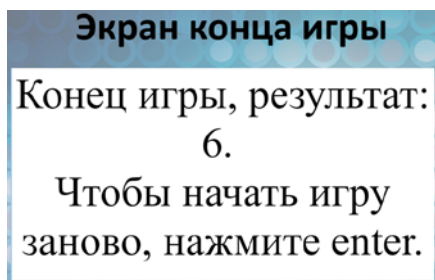


13. Чтобы организовать эту проверку, в функции `cntdwn` добавим `if`, который будет проверять значение нашей переменной на 0. Если она равна нулю, то тогда мы:

- Останавливаем таймер
- Выключаем обработчик передвижения мыши
- Выводим надпись о конце игры на экран
- Прячем все остальные блоки
- Подключаем обработчик нажатия клавиши клавиатуры.



14. Теперь по истечению времени у нас выводится такое окно. Игра официально закончена, можно соревноваться с друзьями.



15. Дополнительные задачи для страждущих.

