SF EDUCATION

Программирование на VBA

Редактирование и изменение ячеек

Содержание:

1. Промежуточные цвета
2. Запрос цвета из ячейки
3. Узоры
4. Формат ячейки
5. Добавление комментария

Промежуточные цвета

Возвращаемся к Excel. Запускаем макрос и видим, что цвет поменялся. Точно также мы можем заменить цвет, например, на красный. Для этого прописываем:

objColorStop.Color = VbRed

Запускаем и видим, что цвет поменялся на красный. Теперь у нас есть полная возможность изменять цвета, которые нам нужны, а с формулой Interior.Gradient.Degree – поворачивать градиент. Таким же способом мы можем вернуть градиент в изначальное положение. Для этого пишем следующее:

Range(“A1”).Interior.Gradient.Degree = 180

Видим, что градиент вернулся в первоначальное положение. Вернём градиент на 90 градусов. Теперь сделаем следующий этап: добавим промежуточные цвета и сделаем их несколько для того, чтобы понять преимущество того, что мы уже сделали. Для этого, добавим в макросе несколько элементов:

- Прописываем, чтобы добавить красный цвет:

Set objColorStops = Range(“A1”).Interior.Gradient.ColorStops.Add(0.33)

objColorStops.Color = vbRed

- Ниже прописываем то же самое, но заменяем цвет на зелёный:

Set objColorStops = Range(“A1”).Interior.Gradient.ColorStops.Add(0.66)

objColorStops.Color = vbGreen

- В самой первой формуле меняем цвет на жёлтый:

Set objColorStops = Range(“A1”).Interior.Gradient.ColorStops.Add(0)

objColorStops.Color = vbYellow

- Последний цвет меняем на синий:

Set objColorStops = Range(“A1”).Interior.Gradient.ColorStops.Add(1)

objColorStops.Color = vbBlue

В макросе должно получиться следующее:

```
Set objColorStop = Range("A1").Interior.Gradient.ColorStops.Add(0)
objColorStop.Color = vbYellow

Set objColorStop = Range("A1").Interior.Gradient.ColorStops.Add(0.33)
objColorStop.Color = vbRed

Set objColorStop = Range("A1").Interior.Gradient.ColorStops.Add(0.66)
objColorStop.Color = vbGreen

Set objColorStop = Range("A1").Interior.Gradient.ColorStops.Add(1)
objColorStop.Color = vbBlue
```

Таким образом, у нас сначала будет градиент из жёлтого в красный, из красного в зелёный и из зелёного в синий. Запускаем и видим, что получился соответствующий градиент. То есть, нулевая позиция у нас была ярко-жёлтым цветом, 0.33 – стандартным красным, 0.66 – стандартным зелёным, а единица обозначена стандартным синим. Таким образом, мы можем сюда впихнуть огромное количество цветов, которые нам нужны и повернуть градиент по желанию.

Запрос цвета из ячейки

Здесь мы затронули элемент получения цвета с определенной ячейки. Разберём подробно, как это сделать. Удаляем строку с градиентом, с ним мы пока что закончили. Удаляем весь текст из макроса.

Если у вас есть таблица с какими-то специфическими цветами и вы не знаете, что это за цвета, **можно запросить цвет из определенной ячейки**.

Закрасим небольшую таблицу (A1:E9) в оранжевый цвет и запросим цвет из ячейки. Для этого:

- Обращаемся к ячейке (Cells), прописываем:

Cells(1,2) – Range(“A1”).Interior.Color

Таким образом, мы запишем в ячейку **Cells(1,2)** значение из ячейки A1. Для наглядности убираем закрашенную таблицу и оставляем закрашенной только ячейку A1. Запускаем макрос и получаем значение в соседней ячейке. Мы получили **Color Index**, то есть **индекс конкретного цвета по внутренней системе Excel**. Попробуем поставить другой цвет. Запускаем макрос и видим индекс этого цвета.

Узоры

Далее рассмотрим узоры – **ColorPatterns**. В Excel есть большое количество различных узоров, рассмотрим несколько типов узоров. Удаляем все ячейки и текст из макроса. В ячейку A1 поместим определённый узор, для этого пишем:

Range(“A1”).Interior.Pattern=xlPattern.xlPatternChecker

Запускаем и видим, что в ячейке появился необходимый узор. Более того, мы можем ещё и закрасить этот узор. Для этого пишем:

Range(“A1”).Interior.PatternColor=vbBlue

Запускаем и видим, что теперь наш узор окрасился в синий. Теперь мы разобрались с Interior, то есть с наполнением ячейки.

Формат ячейки

Двигаемся дальше и рассматриваем формат ячейки:

- Выбираем ячейку;
- Выставляем у неё границы;
- Пишем любой текст и делаем его полужирным;
- Прописываем:

Range(“C3”).ClearFormats

Запускаем и видим, что и текст, и граница были сброшены до стандартных настроек. Это ещё один метод, который мы рассмотрели.

Добавление комментариев

Дополнительный метод, который хотелось бы рассмотреть – это добавление комментариев. Очень часто нам необходимо добавить комментарий в определённую ячейку. Стоит отметить, что VBA **сам предлагает правильный метод** написания комментария. Чтобы добавить комментарий, пишем следующее:

Range(“C3”).AddComment “Комментарий”

Запускаем и видим, что к выбранной ячейке появился комментарий. При наведении курсора на соответствующую ячейку, появляется комментарий. **Таким**

образом мы можем даже автоматически расставлять комментарии к ячейкам там, где нам необходимо.