SF EDUCATION

Программирование на VBA

Редактирование и изменение ячеек

Содержание:

1. Характеристика Fontstyle
2. Характеристика TintAndShade
3. Блок Borders (границы)

Характеристика Fontstyle

Последнее, что мы рассмотрим в блоке Font – это характеристики **Fontstyle** и **TintAndShade**. Fontstyle отвечает за стиль текста, а TintAndShade – за осветление и затемнение. Проверим, как это работает:

- Открываем Excel;
- Открываем Visual Basic;
- Записываем выбранную ячейку с помощью формулы:

```
Range(“B3”).Select
```

- Далее записываем, что мы будем делать с этой ячейкой с помощью формулы:

```
Selection.Font.FontStyle = “Bold Italic“
```

Мы обращаемся к блоку **Font**, а затем к блоку **FontStyle**. Мы можем использовать жирный, курсивный или подчеркнутый текст **одновременно**, то есть не писать несколько строчек, а сразу записать Bold и Italic. Теперь пишем в ячейке какой-нибудь текст и запускаем макрос.

Видим, что текст стал полужирным и курсивным, соответственно, мы можем **в одной строчке использовать несколько характеристик**. За это отвечает FontStyle. Что за что же отвечает следующая характеристика, TintAndShade?

Характеристика TintAndShade

TintAndShade отвечает за осветление и затемнение.

Посмотрим, как это выглядит:

- Сначала, уберём все характеристики текста, для этого, прописываем формулу:

Selection.Font.FontStyle = None

- Далее прописываем следующую формулу:

Selection.Font.TintAndShade

TintAndShade бывает всего трёх видов: -1 (затемнение); 0 (нейтральный); 1 (осветление).

- Выбираем ещё две ячейки (B4 и B5), по аналогии с B3, добавляем текст в новые ячейки;
- Для ячейки B3 ставим значение -1, для B4 – 0, для B5 – 1 должно получиться следующее:

```
Sub text()
Range("B3").Select
Selection.Font.TintAndShade = -1
Range("B4").Select
Selection.Font.TintAndShade = 0
Range("B5").Select
Selection.Font.TintAndShade = 1
End Sub
```

- Для примера окрашиваем текст во всех ячейках в оранжевый;
- Запускаем макрос и видим, что текст в ячейке B3 затемнился и стал чёрным, в ячейке B4 – остался без изменения, а в ячейке B5 – осветлился и стал белым;

Для чего это нужно? Для того, чтобы, когда вы получаете таблицы разного формата, где каждая ячейка может быть с разным содержанием, с разной коррекцией, у текста своей шрифт, цвет и стиль, можно было сбросить все настройки ячеек. Тем самым, не будет необходимости вручную исправлять весь текст. С характеристиками блока Font мы на сегодня закончили.

Блок Borders (границы)

Переходим к блоку **Borders**. Первое, что мы будет делать в Borders – это рисовать границы. Всего существует восемь типов Borders, то есть не характеристики ячейки, а самого типа блока Borders. Сначала мы будем обращаться к Bordersy, к границам, потом выбирать тип, а после этого, выбирать, что делать с этой границей. Для начала рассмотрим границы следующего содержания:

- Нарисовать левую границу;
- Верхнюю границу;
- Нижнюю границу;

- Правую границу;
- Все внутренние вертикальные границы;
- Все внутренние горизонтальные границы;
- Все диагональные границы по одним углом;
- Все диагональные границы под другим углом.

Перейдём в Excel и постараемся всё это сделать:

- Заходим в Visual Basic;
- Выбираем девять ячеек, в которых мы будем всё это наблюдать, сделаем таблицу с B3 по D5, прописываем **Range(“B3:D5”).Select**;
- Далее, к эти выбранным ячейкам будем обращаться при помощи Borders, для этого прописываем Selection.Borders;
- Для начала нарисуем левую границу, прописываем формулу:

Selection.Borders (xlEdgeLeft)

Мы обратились к типу Borders, то есть к границам, а конкретно к самой левой в выбранном диапазоне. После этого, мы обращаемся к линии, которую будем рисовать, и дописываем:

Selection.Borders (xlEdgeLeft) **.LineStyle=xlContinuous**

Запускаем макрос. Видим, что для выделенных ячеек была нарисована самая левая граница. Поступим точно также и построим верхнюю границу. **Заменяем слово «Left» на слово «Top»**, остальное оставляем без изменений, и получаем:

Selection.Borders (xlEdgeTop) .LineStyle=xlContinuous

Снова запускаем макрос, убираем выделение таблицы и видим, что появилась верхняя граница. Делаем то же самое для нижней границы. Записываем:

Selection.Borders (xlEdgeBottom) .LineStyle=xlContinuous

В финале нам осталось сделать только правую границу. Записываем:

Selection.Borders (xlEdgeRight) .LineStyle=xlContinuous

Запускаем и видим, что наша таблицы была выделена благодаря этим четырём функциям по краям. А теперь подумаем, как выделить все прямые внутри таблицы? Для этого убираем **xlEdgeRight**. Сначала пропишем внутри только вертикальные границы. Записываем

Selection.Borders (xlInsideVertical) .LineStyle=xlContinuous

Запускаем и видим, что внутри нашей таблицы нарисовались вертикальные линии. Возникает вопрос, будут ли выделяться внешние вертикальные линии?

Чтобы проверить, выбираем новую таблицу с B7 и D9 и используем с ней ту же формулу. Видим, что выделяются только внутренние вертикальные линии, внешние – нет. Аналогично делаем горизонтальные линии. Записываем:

Selection.Borders (xlInsideHorizontal) .LineStyle=xlContinuous

Запускаем и видим, что в таблице нарисовались и горизонтальные линии.

Далее, завершающий момент – рисуем диагональные линии. Изменим таблицу на B11 – D13 и прописываем диагональные линии. Для этого записываем:

Selection.Borders (xlInsideDiagonalDown) .LineStyle=xlContinuous

У Diagonal есть два типа: Down и Up. Для начала рассмотрим типа Down

Запускаем и видим, что появились линии, идущие **от верхнего правого угла к нижнему левому**. Далее, заменяем Down на Up:

Selection.Borders (xlInsideDiagonalUp) .LineStyle=xlContinuous

Запускаем и видим, что поверх предыдущих диагоналей появились новые **от верхнего левого угла к нижнему правому**. Мы разобрались как рисовать границы. Переходим к следующему этапу. Мы продолжаем работать с характеристиками Borders. Всего есть **шесть типов границ**. Один тип (xlContinuous) мы рассмотрели. Далее мы рассмотрим оставшиеся типы границы и **толщины границ, их всего три**. Кажется, для чего это нужно, если можно вручную выделить определенный диапазон ячеек и сделать так, как нужно?

Вручную мы делаем это каждый раз и тратим время, а машинный код служит для **автоматизации процесса**.

Таким образом, у нас изначально будет собираться таблицы и можно будет обрисовать её так, как нужно, выделить определённые ячейки и даже отправить собранную таблицу по почте. Переходим в Excel и пробуем это всё разобрать:

- Выбираем новую таблицу F2:H2, нам достаточно несколько ячеек, потому что мы не будет ничего рисовать внутри таблицы, а просто сделаем границы;
- Оставляем xlEdgeBottom, так как мы будем рисовать нижнюю границу таблицы;
- Выбираем стиль xlDot;
- Запускаем макрос;
- Снимаем выделение и видим, что нижняя граница выделилась пунктирной линией (в точку).

Переходим к стилю **xlDashDot**, для этого выберем ячейки **F3:H3**, запускам макрос и видим новый вид пунктирной линии на границе (точки и тире).

Следующий тип пунктирной линии это **xlDash**. Выбираем новую таблицу **F4:H4**. Запускаем, снимаем выделение и видим новый тип пунктирной линии. Каждый тип пунктирной линии уникален, поэтому важно понимать, какой именно тип пунктирной линии нам нужен. Для того, чтобы открыть подсказку по видам пунктирных линий, прописываем **xl** и нажимаем **CTRL+пробел**. Откроется список, из которого можно выбрать нужны тип.

Далее рассмотрим тип **xlSlantDashDot** для ячеек **F5:H5**. Запускаем макрос, снимаем выделение и видим новый тип границы.

Следующий тип – это **xlDouble**, который представляет собой двойную границу. Выбираем ячейки **F6:H6**. Запускаем макрос, снимаем выделение и видим новый тип границы. Это шесть основных типов границ. Их можно комбинировать между собой.

Далее мы переходим к толщине линий.

