

Сложность \*

### 3.1. Задание \*

Все задания выполняйте в модуле управляемого приложения, в процедуре ПриНачалеРаботыСистемы(). После того как закончите, удалите их из модуля, чтобы они не мешали вам в дальнейшем.

Создайте переменную МойРост. Запишите в нее свой рост. Затем измените значение этой переменной на тот рост, который будет у вас, например, через год.

Запустите «1С:Предприятие» в режиме отладки и посмотрите, как будет изменяться значение этой переменной.

### 3.2. Задание \*\*

Создайте переменную, в которой будет храниться ваше имя. Посмотрите значение этой переменной в режиме отладки.

### 3.3. Задание \*\*

Создайте переменную, в которой будет храниться домашний адрес. Ваш домашний адрес или адрес другого человека - это не важно. Запишите в нее свой адрес.

### 3.4. Задание \*\*

Выберите любой учебный день. Создайте две переменные. В одной будет храниться название первого урока в этот день, в другой - название второго урока.

### 3.5. Задание \*\*

Создайте переменную, в которой будет храниться телефонный номер. Ваш номер или любого другого человека - это не важно. Запишите в нее свой телефонный номер.

### 3.6. Задание \*\*

Создайте переменную, в которой будет храниться оценка, полученная на уроке. На одном уроке вы получили пятерку, на другом уроке - четыре с плюсом. Запишите это в программе. Посмотрите в режиме отладки, как будет изменяться значение этой переменной.

### 3.7. Задание \*

В одной переменной сохраните вашу среднюю скорость - 5 км/ч. В другой переменной сохраните расстояние до школы - 6 км. В третьей переменной посчитайте количество минут, которое вам понадобится, чтобы дойти до школы.

### 3.8. Задание \*\*

Используйте среднюю скорость и расстояние до школы из задания 3.7. В третьей переменной посчитайте, сколько раз за сутки вы сможете дойти до школы и вернуться обратно, если не будете останавливаться и спать. После этого с помощью вычисления выражения посмотрите, сколько километров вы можете пройти за сутки.

### 3.9. Задание \*\*

В этом же примере (3.8) посчитайте, сколько раз вы сможете сделать то же самое, если будете ехать на велосипеде только в светлое время суток. Средняя скорость велосипедиста - 15 км/ч. Светлое время суток длится в среднем 13 часов.

### 3.10. Задание \*\*

В вашем портфеле были только учебники. В понедельник, чтобы перекусить в школе, вы взяли из дома 2 яблока. Но не стали их есть, и они остались в портфеле.

Во вторник и в среду вы тоже брали яблоки из дома и оставляли их в портфеле.

Сколько яблок будет в вашем портфеле в среду, если каждый день вы брали из дома на два яблока больше, чем в предыдущий? Для решения этой задачи используйте две переменные: `ВПортфеле` (количество яблок в портфеле) и `ВзялИзДома` (количество яблок, которое вы взяли из дома).

### 3.11. Задание \*

Создайте четыре переменные:

- `ДлинаУрока` - 45 минут;
- `ДлинаПеремены` - 15 минут;
- `ДлинаБольшойПеремены` - 25 минут;
- `ВсегоУроков` - 6.

Посчитайте, сколько минут вы проводите в школе в течение дня, если одна из перемен между уроками всегда большая.

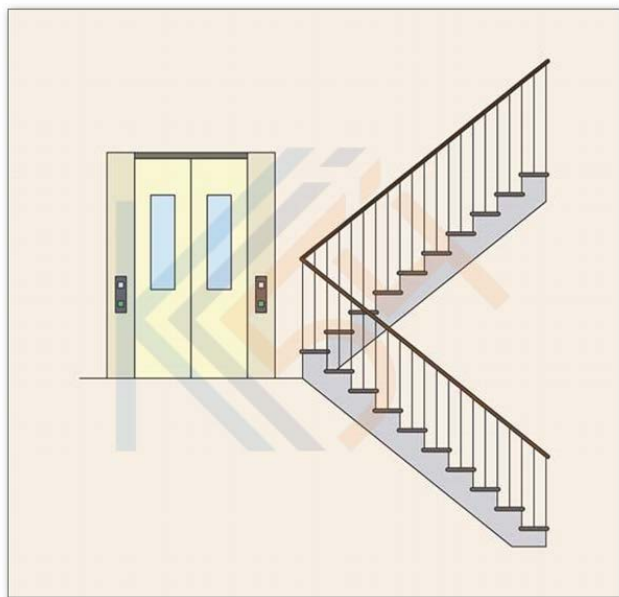


Рис. 3.57. Лестница в многоквартирном доме

- количество ступеней перед входом в подъезд;
- количество ступеней в марше;
- этаж.

### 3.13. Задание \*

Используя переменную Возраст типа Строка, запишите фразу «Мой возраст 21 год».

### 3.14. Задание \*\*

При отправке почтовых сообщений существует определенный порядок перечисления реквизитов адреса:

1. Название улицы, номер дома, номер квартиры.
2. Название населенного пункта (города, поселка и т. п.).
3. Название района.
4. Название республики, края, области, автономного округа (области).

Используя переменные Улица, НомерДома, НомерКвартиры, Гзрод и Область, запишите адрес, выделенный на рисунке 3.60.

Рис. 3.60. Адрес отправителя

### 3.15. Задание \*

Запишите начало дня 2 сентября 2016 года с помощью литерала даты.

### 3.16. Задание \*

Запишите начало дня 2 сентября 2016 года с помощью функции Дата().

### 3.17. Задание \*

В одной переменной сохраните произвольную дату. В другой переменной вычислите начало следующего понедельника для произвольной даты.

### 3.18. Задание \*\*

В одной переменной сохраните произвольную дату. В другой переменной вычислите девять утра для произвольной даты.

### 3.19. Задание \*\*

Разность двух дат представьте в виде количества часов, минут и секунд. Например, «3 ч. 29 мин. 40 с».

### 3.20. Задание \*

Чтобы по пути на работу защититься от дождя, вы берете с собой зонт. С помощью переменных ЯИдуНаРаботу и ИдетДождь вычислите значение переменной НадоВзятьЗонт.

Проверьте, что ваша инструкция правильно работает при любых значениях исходных переменных.

### 3.21. Задание \*

В каких случаях вы не ходите в школу? Когда выходной (праздники, каникулы) и когда вы болеете. С помощью переменных СегодняВыходной и ЯБолею вычислите значение переменной ЯНеИдуВШколу.

Проверьте, что ваша инструкция правильно работает при любых значениях исходных переменных.

### 3.22. Задание \*\*

В хорошую погоду, когда у вас нет занятий, вы всегда идете гулять. С помощью переменных ХорошаяПогода, СегодняВыходной и СегодняПраздник вычислите значение переменной ЯИдуГулять.

Проверьте, что ваша инструкция правильно работает при любых значениях исходных переменных.

### 3.23. Задание \*\*

Вы работаете только по будним дням. С помощью переменных СегодняСуббота и СегодняВоскресенье вычислите значение переменной ЯИдуНаРаботу.

Проверьте, что ваша инструкция правильно работает при любых значениях исходных переменных.

### 3.24. Задание \*

Вы с другом каждый день играете в игру через Интернет. В будний день родители разрешают вам играть только один час. В выходные дни вы можете играть по 4 часа. Используя переменную НомерДняНедели, вычислите, сколько часов вы можете играть с другом в выбранный день. Результат поместите в переменную ВремяДляИгры.

### 3.25. Задание \*\*

В задании 3.24 нужно учесть, что номер дня недели может быть задан с ошибкой. В случае ошибки ваша программа должна вернуть результат 0.

### 3.26. Задание \*\*

Супермаркет работает с 9 часов утра до 8 часов вечера. Вам нужно сходить в супермаркет и купить кефир. Если кефира не будет, то ряженку. Если не будет ни того, ни другого, тогда нужно зайти в круглосуточный магазин и купить молоко, если оно там есть.

Используйте переменные ЕстьКефир, ЕстьРяженка, ЕстьМолоко и ТекущийЧас.

Название своей покупки в виде строки поместите в переменную МояПокупка.

### 3.27. Задание \*

Выведите количество дней в каждом месяце из второй декады в виде «4-й месяц 30 дней». Используйте переменную НомерМесяца. Результат помещайте в переменную Сказать. С помощью пошагового выполнения проверьте все значения, которые окажутся в переменной Сказать.

### 3.28. Задание \*\*

Перечислите школьные оценки от самой лучшей к самой худшей. Результат помещайте в переменную Сказать.

### 3.29. Задание \*

Создайте функцию, которая получает значение типа Дата. А возвращает эта функция текстовое представление месяца и года этой даты.

Для формирования текстового представления используйте функцию ПредставлениеПериода(). Ее описание вы найдете в синтакс-помощнике в ветке Глобальный контекст — Функции форматирования.

### 3.30. Задание \*\*

Возьмите программу из задания 3.19. Преобразуйте ее в функцию, которая получает две даты, а возвращает их разность в виде количества часов, минут и секунд.

### 3.31. Задание \*

Пример из задания 3.29 преобразуйте в процедуру. Текстовое представление месяца и года переданной даты показывайте пользователю с помощью функции ПоказатьОповещениеПользователя().

### 3.32. Задание \*

Пример из задания 3.30 преобразуйте в процедуру.

Представление периода показывайте пользователю с помощью функции ПоказатьОповещениеПользователя().

### 3.33. Задание \*

Создайте массив и запишите в него названия всех месяцев по порядку.

### 3.34. Задание \*

Посмотрите содержимое массива из предыдущего задания в конфигураторе.

### 3.35. Задание \*

В массив из задания 3.33 добавьте два элемента.

Один элемент, «— Начало лета», добавьте перед месяцем, который называется «Июнь». Этот месяц нужно найти по названию.

Второй элемент, «— Конец лета», добавьте после месяца, который называется «Август». Этот месяц тоже нужно найти по названию.

Посмотрите в конфигураторе, правильно ли вы выделили летние месяцы.



### 3.36. Задание \*\*

С помощью цикла обойдите все элементы массива и добавьте в конец каждого названия текущий год. Чтобы, например, вместо «Январь» стало «Январь 2016 г.».

Посмотрите в конфигураторе, правильно ли выглядит результат.

### 3.37. Задание \*

Вы ведете список учеников, которые поедут на экскурсию. Список постоянно меняется, кто-то добавляется, кто-то вычеркивается. Текущий состав списка вы не знаете.

Вам сообщили, что ученик Захаров заболел, и его нужно вычеркнуть из списка.

Создайте массив и заполните его фамилиями Сергеев, Дмитриев, Захаров, Максимов.

Напишите программу, которая удаляет из списка ученика по фамилии Захаров.

Теперь в инструкциях, которые заполняют массив фамилиями, замените фамилию Захаров на фамилию Александров. Проверьте, что ваша программа по-прежнему работает без ошибок.

### 3.38. Задание \*

Создайте массив и заполните его названиями дней недели по порядку. С помощью операции [] получите названия третьего и пятого дней недели.

### 3.39. Задание \*

Создайте массив и заполните его названиями дней недели по порядку. С помощью операции [] к выходным дням допишите «вых.». Чтобы, например, вместо «Суббота» получилось «Суббота вых.».

### 3.40. Задание \*

Используйте массив ДниНедели, показанный на рисунке 3.206. Добавьте к нему листинг 3.9.

Теперь представьте, что для будних дней вам нужно не просто выполнить одну строку БудниеДниДобавить(ТекущийЭлемент), а реализовать в этом месте большой и сложный алгоритм. Прятать такой алгоритм внутри условия Если нехорошо и неудобно.

В этом случае обычно поступают следующим образом. В начале цикла сразу же определяют, подходит очередной день для выполнения алгоритма или нет. Если не подходит (выходной), то просто переходят к следующей итерации цикла. Больше никаких проверок не делают. В результате получается, что инструкции, написанные после Если, будут выполняться только для подходящих элементов.

Измените листинг 3.9 так, чтобы он соответствовал этой стратегии. С помощью пошаговой отладки посмотрите, как работает цикл.

### 3.41. Задание \*

Используйте массив ДниНедели, показанный на рисунке 3.206. Добавьте к нему листинг 3.9.

Обратная ситуация. Алгоритм, который нужно выполнить для будних дней, небольшой и несложный.

В этом случае с помощью инструкции Если отбирают только те элементы, которые подходят для этого алгоритма. И выполняют алгоритм. Получается, что после Если нет никаких инструкций, и для неподходящих элементов ничего не выполняется.

Измените листинг 3.9 так, чтобы он соответствовал этой стратегии. С помощью пошаговой отладки посмотрите, как работает цикл.

### 3.42. Задание \*

Используйте массив ДниНедели, показанный на рисунке 3.206. Добавьте к нему листинг 3.8, в котором используется инструкция Для По Цикл.

Дни недели расположены в массиве в правильном порядке. Вы обходите их в цикле в том же правильном порядке. Вы точно знаете, что после субботы будет воскресенье, которое вам не нужно.

Измените листинг так, чтобы не анализировались те дни, которые вам заведомо не нужны. С помощью пошаговой отладки посмотрите, как работает цикл.

### 3.43. Задание \*\*

Создайте массив и в цикле заполните его годами с 2000 по 2020. Затем из этого массива отберите только високосные годы и поместите их в другой массив.

Високосным считается год, который делится на 4 без остатка.

Объясните, почему вы использовали ту или иную инструкцию цикла. Посмотрите в конфигураторе, сколько високосных лет у вас получилось.

### 3.44. Задание \*

Создайте и заполните структуру, в которой будет храниться ваш домашний адрес: улица, номер дома, корпус, квартира, город, район, область, край, республика.

Создайте в структуре только те элементы, которые есть в вашем адресе.

### 3.45. Задание \*

В конфигураторе посмотрите значения отдельных элементов вашей структуры.



Например, улица и город.

### 3.46. Задание \*

Используйте структуру из задания 3.44. В одной переменной сформируйте адрес вашего населенного пункта, а в другой - ваш адрес в этом населенном пункте.

### 3.47. Задание \*

Создайте структуру, в которой будут содержаться дни рождения двоих ваших самых близких друзей или родственников. Создайте и заполните эту структуру только с помощью конструктора.

### 3.48. Задание \*

Структура адреса из примера 3.46 может содержать корпус дома, а может и не содержать его. Сформируйте вторую переменную этого примера (ваш адрес в населенном пункте) так, чтобы программа работала без ошибок как в одном, так и в другом случае.

### 3.49. Задание \*

В каждом документе УчебныйДень удалите последний урок. В каждом учебном дне должно остаться по 5 уроков.

Для выполнения этого задания создайте новую процедуру в общем модуле Серверный. Перед запуском и проверкой своей программы запишите где-нибудь, какой урок должен оказаться последним в каждый из дней. Чтобы у вас была возможность проверить правильность работы.

### 3.50. Задание \*\*

Получите документ за 2 сентября 2021 г. и удалите в нем первый урок. Для этого самостоятельно найдите в синтакс-помощнике метод, который позволяет найти документ по номеру. Обработайте ситуацию, что документа с таким номером может и не быть в базе.

Перед выполнением запомните, какой урок был первым, чтобы потом проверить.

После проверки в тексте программы измените номер документа на несуществующий. Проверьте, как работает ваша программа в случае, когда искомый документ отсутствует.

Подсказка 1. Чтобы определить, является ли ссылка на документ пустой, у типа ДокументСсылка.<Имя документа> существует метод Пустая().

Подсказка 2. Чтобы, находясь на сервере, сообщить что-нибудь клиенту, используйте объект СообщениеПользователю. Самостоятельно найдите его описание в синтакс-помощнике. Вам понадобится создать этот объект, заполнить свойство Текст и выполнить метод Сообщить().

### 3.51. Задание \*\*

Создайте документ 8 сентября 2021 г. и запишите в него два урока информатики.

Самостоятельно найдите метод, который позволяет создать документ.

Подсказка 1. У документа нужно обязательно заполнить дату.

Подсказка 2. Способы работы со справочниками такие же, как и с документами. В справочнике Предметы название предмета хранится в поле Код.

Подсказка 3. Если вы все сделали правильно, после запуска приложения вы не увидите свой новый документ в списке. Это связано с тем, что вы делаете учебный пример в обработчике ПриНачалеРаботыСистемы(). Но ваш новый документ есть в списке. Чтобы его увидеть, выполните команду Еще – Обновить.

