

## Acadgild Solutions for Business analysis with R \_session 4 \_Assignment 1

Solutions for Question 1 to Question 6

Question 1 Create the vectors

(a) (2, 3, ..., 29, 30)

(b) (30, 29, ..., 2)

(c) (1, 2, 3, ..., 29, 30, 29, 28, ..., 2, 1)

(d) (4, 6, 3) and assign it to the name dev.

For parts (e), (f) and (g) .

(e) (5, 6, 7, 5, 6, 7, , 5, 6, 7) where there are 10 occurrences of 5.

(f) (5, 6, 7, 5, 6, 7, , 5, 6, 7, 5) where there are 11 occurrences of 5, 10 occurrences of 6 and 10 occurrences of 7.

(g) (4, 4, , 4, 6, 6, , 6, 3, 3, , 3) where there are 10 occurrences of 4, 20 occurrences of 6 and 30 occurrences of 3.

### Answer 1

#### RCommands:

Part a)

a<-(2:30)

a

```
5:2 (Top Level) R Script
Console ~/
> a<-(2:30)
> a
[1] 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
[24] 25 26 27 28 29 30
> |
```

#### Part b)

b<-(30:2)

b

```
Console ~/
> b<-(30:2)
> b
[1] 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8
[24] 7 6 5 4 3 2
> |
```

#### Part c)

c<-c(1:30,29:1)

c

```
Console ~/
> c<-c(1:30,29:1)
> c
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
[24] 24 25 26 27 28 29 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14
[47] 13 12 11 10 9 8 7 6 5 4 3 2 1
> |
```

#### Part d)

dev<- c(4,6,3)

dev

```
Console ~/
> dev<- c(4,6,3)
> dev
[1] 4 6 3
> |
```



```

Console ~
> x<-c(seq(3.1,6,0.1))
> x
[1] 3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8 3.9 4.0 4.1 4.2 4.3 4.4 4.5 4.6 4.7 4.8 4.9
[20] 5.0 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9 6.0
> y<-sin(x)
> y
[1] 0.04158066 -0.05837414 -0.15774569 -0.25554110 -0.35078323 -0.44252044
[7] -0.52983614 -0.61185789 -0.68776616 -0.75680250 -0.81827711 -0.87157577
[13] -0.91616594 -0.95160207 -0.97753012 -0.99369100 -0.99992326 -0.99616461
[19] -0.98245261 -0.95892427 -0.92581468 -0.88345466 -0.83226744 -0.77276449
[25] -0.70554033 -0.63126664 -0.55068554 -0.46460218 -0.37387666 -0.27941550
> p<-exp(x)
> p
[1] 22.19795 24.53253 27.11264 29.96410 33.11545 36.59823 40.44730
[8] 44.70118 49.40245 54.59815 60.34029 66.68633 73.69979 81.45087
[15] 90.01713 99.48432 109.94717 121.51042 134.28978 148.41316 164.02191
[22] 181.27224 200.33681 221.40642 244.69193 270.42641 298.86740 330.29956
[29] 365.03747 403.42879
> r<-c(y*p)
> r
[1] 0.9230055 -1.4320654 -4.2769020 -7.6570591 -11.6163451
[6] -16.1954669 -21.4304437 -27.3507725 -33.9773327 -41.3200162
[11] -49.3750762 -58.1221905 -67.5212405 -77.5088155 -87.9944570
[16] -98.8566695 -109.9387348 -121.0443775 -131.9333449 -142.3169809
[21] -151.8538900 -160.1458060 -166.7338044 -171.0950158 -172.6400256
[26] -170.7111690 -164.5819569 -153.4578954 -136.4789910 -112.7242573
>

```

**Question 3 : Execute the following lines which create two vectors of random integers which are chosen with replacement from the integers 0, 1, ..., 999. Both vectors have length 250.**

- ```
set.seed(100)
x <- Sample(0:999, 250, replace=T)
y <- Sample(0:999, 250, replace=T)
```
- Identify out the values in y which are > 500.
  - Identify the index positions in y of the values which are > 700?
  - What are the values in x which are in Same index position to the values in y which are > 400?
  - How many values in y are within 200 of the maximum value of the terms in y?
  - How many numbers in x are divisible by 2?
  - Sort the numbers in the vector x in the order of increasing values in y.
  - Create the vector  $(x_1 + 2x_2 - x_3; x_2 + 2x_3 - x_4, \dots, x_{n-2} + 2x_{n-1} - x_n)$ .

**Answer 3**

**Rcommands:**

```
set.seed(100)
x <- Sample(0:999, 250, replace=T)
y <- Sample(0:999, 250, replace=T)
```

**Part a)**

```
y[y>500]
```

**Part b)**

```

> y[y>700]
701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739
740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778
779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817
818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856
857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895
896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934
935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973
974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999
>

```

### Part c)

```

> x[y>400] #x[y>400] shows x values in the same index of y
401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439
440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478
479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517
518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556
557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595
596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634
635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673
674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712
713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751
752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790
791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829
830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868
869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907
908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946
947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985
986 987 988 989 990 991 992 993 994 995 996 997 998 999
>

```

### Part d)

```
Console ~/ / 
> sum(y>max(y)-200)
[1] 200
> |
```



### Part e)

```
length(y[y%%2==0])
```

```
61:1 (Top Level)
Console ~/
> length(y[y%%2==0])
[1] 501
```

### Part f)

```
x[order(y)]
```

```
Console ~/ > x[order(y)]
```

|     | replace |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|-----|---------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   | 1       | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  | 11  | 12  | 13  | 14  | 15  | 16  |
| 17  | 18      | 19  | 20  | 21  | 22  | 23  | 24  | 25  | 26  | 27  | 28  | 29  | 30  | 31  | 32  | 33  | 34  |
| 35  | 36      | 37  | 38  | 39  | 40  | 41  | 42  | 43  | 44  | 45  | 46  | 47  | 48  | 49  | 50  | 51  | 52  |
| 53  | 54      | 55  | 56  | 57  | 58  | 59  | 60  | 61  | 62  | 63  | 64  | 65  | 66  | 67  | 68  | 69  | 70  |
| 71  | 72      | 73  | 74  | 75  | 76  | 77  | 78  | 79  | 80  | 81  | 82  | 83  | 84  | 85  | 86  | 87  | 88  |
| 89  | 90      | 91  | 92  | 93  | 94  | 95  | 96  | 97  | 98  | 99  | 100 | 101 | 102 | 103 | 104 | 105 | 106 |
| 107 | 108     | 109 | 110 | 111 | 112 | 113 | 114 | 115 | 116 | 117 | 118 | 119 | 120 | 121 | 122 | 123 | 124 |
| 125 | 126     | 127 | 128 | 129 | 130 | 131 | 132 | 133 | 134 | 135 | 136 | 137 | 138 | 139 | 140 | 141 | 142 |
| 143 | 144     | 145 | 146 | 147 | 148 | 149 | 150 | 151 | 152 | 153 | 154 | 155 | 156 | 157 | 158 | 159 | 160 |
| 161 | 162     | 163 | 164 | 165 | 166 | 167 | 168 | 169 | 170 | 171 | 172 | 173 | 174 | 175 | 176 | 177 | 178 |
| 179 | 180     | 181 | 182 | 183 | 184 | 185 | 186 | 187 | 188 | 189 | 190 | 191 | 192 | 193 | 194 | 195 | 196 |
| 197 | 198     | 199 | 200 | 201 | 202 | 203 | 204 | 205 | 206 | 207 | 208 | 209 | 210 | 211 | 212 | 213 | 214 |
| 215 | 216     | 217 | 218 | 219 | 220 | 221 | 222 | 223 | 224 | 225 | 226 | 227 | 228 | 229 | 230 | 231 | 232 |
| 233 | 234     | 235 | 236 | 237 | 238 | 239 | 240 | 241 | 242 | 243 | 244 | 245 | 246 | 247 | 248 | 249 | 250 |
| 250 | 251     | 252 | 253 | 254 | 255 | 256 | 257 | 258 | 259 | 260 | 261 | 262 | 263 | 264 | 265 | 266 | 267 |
| 268 | 269     | 270 | 271 | 272 | 273 | 274 | 275 | 276 | 277 | 278 | 279 | 280 | 281 | 282 | 283 | 284 | 285 |
| 286 | 287     | 288 | 289 | 290 | 291 | 292 | 293 | 294 | 295 | 296 | 297 | 298 | 299 | 300 | 301 | 302 | 303 |

```
Console ~/
61:1 (Top Level)
```

|     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 304 | 305 | 306 | 307 | 308 | 309 | 310 | 311 | 312 | 313 | 314 | 315 | 316 | 317 | 318 | 319 | 320 | 321 |
| 322 | 323 | 324 | 325 | 326 | 327 | 328 | 329 | 330 | 331 | 332 | 333 | 334 | 335 | 336 | 337 | 338 | 339 |
| 340 | 341 | 342 | 343 | 344 | 345 | 346 | 347 | 348 | 349 | 350 | 351 | 352 | 353 | 354 | 355 | 356 | 357 |
| 358 | 359 | 360 | 361 | 362 | 363 | 364 | 365 | 366 | 367 | 368 | 369 | 370 | 371 | 372 | 373 | 374 | 375 |
| 376 | 377 | 378 | 379 | 380 | 381 | 382 | 383 | 384 | 385 | 386 | 387 | 388 | 389 | 390 | 391 | 392 | 393 |
| 394 | 395 | 396 | 397 | 398 | 399 | 400 | 401 | 402 | 403 | 404 | 405 | 406 | 407 | 408 | 409 | 410 | 411 |
| 412 | 413 | 414 | 415 | 416 | 417 | 418 | 419 | 420 | 421 | 422 | 423 | 424 | 425 | 426 | 427 | 428 | 429 |
| 430 | 431 | 432 | 433 | 434 | 435 | 436 | 437 | 438 | 439 | 440 | 441 | 442 | 443 | 444 | 445 | 446 | 447 |
| 448 | 449 | 450 | 451 | 452 | 453 | 454 | 455 | 456 | 457 | 458 | 459 | 460 | 461 | 462 | 463 | 464 | 465 |
| 466 | 467 | 468 | 469 | 470 | 471 | 472 | 473 | 474 | 475 | 476 | 477 | 478 | 479 | 480 | 481 | 482 | 483 |
| 484 | 485 | 486 | 487 | 488 | 489 | 490 | 491 | 492 | 493 | 494 | 495 | 496 | 497 | 498 | 499 | 500 | 501 |
| 502 | 503 | 504 | 505 | 506 | 507 | 508 | 509 | 510 | 511 | 512 | 513 | 514 | 515 | 516 | 517 | 518 | 519 |
| 520 | 521 | 522 | 523 | 524 | 525 | 526 | 527 | 528 | 529 | 530 | 531 | 532 | 533 | 534 | 535 | 536 | 537 |
| 538 | 539 | 540 | 541 | 542 | 543 | 544 | 545 | 546 | 547 | 548 | 549 | 550 | 551 | 552 | 553 | 554 | 555 |
| 556 | 557 | 558 | 559 | 560 | 561 | 562 | 563 | 564 | 565 | 566 | 567 | 568 | 569 | 570 | 571 | 572 | 573 |
| 574 | 575 | 576 | 577 | 578 | 579 | 580 | 581 | 582 | 583 | 584 | 585 | 586 | 587 | 588 | 589 | 590 | 591 |
| 592 | 593 | 594 | 595 | 596 | 597 | 598 | 599 | 600 | 601 | 602 | 603 | 604 | 605 | 606 | 607 | 608 | 609 |
| 610 | 611 | 612 | 613 | 614 | 615 | 616 | 617 | 618 | 619 | 620 | 621 | 622 | 623 | 624 | 625 | 626 | 627 |

|     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 628 | 629 | 630 | 631 | 632 | 633 | 634 | 635 | 636 | 637 | 638 | 639 | 640 | 641 | 642 | 643 | 644 | 645 |
| 646 | 647 | 648 | 649 | 650 | 651 | 652 | 653 | 654 | 655 | 656 | 657 | 658 | 659 | 660 | 661 | 662 | 663 |
| 664 | 665 | 666 | 667 | 668 | 669 | 670 | 671 | 672 | 673 | 674 | 675 | 676 | 677 | 678 | 679 | 680 | 681 |
| 682 | 683 | 684 | 685 | 686 | 687 | 688 | 689 | 690 | 691 | 692 | 693 | 694 | 695 | 696 | 697 | 698 | 699 |
| 700 | 701 | 702 | 703 | 704 | 705 | 706 | 707 | 708 | 709 | 710 | 711 | 712 | 713 | 714 | 715 | 716 | 717 |
| 718 | 719 | 720 | 721 | 722 | 723 | 724 | 725 | 726 | 727 | 728 | 729 | 730 | 731 | 732 | 733 | 734 | 735 |
| 736 | 737 | 738 | 739 | 740 | 741 | 742 | 743 | 744 | 745 | 746 | 747 | 748 | 749 | 750 | 751 | 752 | 753 |
| 754 | 755 | 756 | 757 | 758 | 759 | 760 | 761 | 762 | 763 | 764 | 765 | 766 | 767 | 768 | 769 | 770 | 771 |
| 772 | 773 | 774 | 775 | 776 | 777 | 778 | 779 | 780 | 781 | 782 | 783 | 784 | 785 | 786 | 787 | 788 | 789 |
| 790 | 791 | 792 | 793 | 794 | 795 | 796 | 797 | 798 | 799 | 800 | 801 | 802 | 803 | 804 | 805 | 806 | 807 |
| 808 | 809 | 810 | 811 | 812 | 813 | 814 | 815 | 816 | 817 | 818 | 819 | 820 | 821 | 822 | 823 | 824 | 825 |
| 826 | 827 | 828 | 829 | 830 | 831 | 832 | 833 | 834 | 835 | 836 | 837 | 838 | 839 | 840 | 841 | 842 | 843 |
| 844 | 845 | 846 | 847 | 848 | 849 | 850 | 851 | 852 | 853 | 854 | 855 | 856 | 857 | 858 | 859 | 860 | 861 |
| 862 | 863 | 864 | 865 | 866 | 867 | 868 | 869 | 870 | 871 | 872 | 873 | 874 | 875 | 876 | 877 | 878 | 879 |
| 880 | 881 | 882 | 883 | 884 | 885 | 886 | 887 | 888 | 889 | 890 | 891 | 892 | 893 | 894 | 895 | 896 | 897 |
| 898 | 899 | 900 | 901 | 902 | 903 | 904 | 905 | 906 | 907 | 908 | 909 | 910 | 911 | 912 | 913 | 914 | 915 |
| 916 | 917 | 918 | 919 | 920 | 921 | 922 | 923 | 924 | 925 | 926 | 927 | 928 | 929 | 930 | 931 | 932 | 933 |
| 934 | 935 | 936 | 937 | 938 | 939 | 940 | 941 | 942 | 943 | 944 | 945 | 946 | 947 | 948 | 949 | 950 | 951 |

|     |     |     |     |     |     |     |     |     |     |                                                         |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---------------------------------------------------------|-----|-----|-----|-----|-----|-----|-----|
| 718 | 719 | 720 | 721 | 722 | 723 | 724 | 725 | 726 | 727 | 728                                                     | 729 | 730 | 731 | 732 | 733 | 734 | 735 |
| 736 | 737 | 738 | 739 | 740 | 741 | 742 | 743 | 744 | 745 | 746                                                     | 747 | 748 | 749 | 750 | 751 | 752 | 753 |
| 754 | 755 | 756 | 757 | 758 | 759 | 760 | 761 | 762 | 763 | 764                                                     | 765 | 766 | 767 | 768 | 769 | 770 | 771 |
| 772 | 773 | 774 | 775 | 776 | 777 | 778 | 779 | 780 | 781 | 782                                                     | 783 | 784 | 785 | 786 | 787 | 788 | 789 |
| 790 | 791 | 792 | 793 | 794 | 795 | 796 | 797 | 798 | 799 | 800                                                     | 801 | 802 | 803 | 804 | 805 | 806 | 807 |
| 808 | 809 | 810 | 811 | 812 | 813 | 814 | 815 | 816 | 817 | 818                                                     | 819 | 820 | 821 | 822 | 823 | 824 | 825 |
| 826 | 827 | 828 | 829 | 830 | 831 | 832 | 833 | 834 | 835 | 836                                                     | 837 | 838 | 839 | 840 | 841 | 842 | 843 |
| 844 | 845 | 846 | 847 | 848 | 849 | 850 | 851 | 852 | 853 | 854                                                     | 855 | 856 | 857 | 858 | 859 | 860 | 861 |
| 862 | 863 | 864 | 865 | 866 | 867 | 868 | 869 | 870 | 871 | 872                                                     | 873 | 874 | 875 | 876 | 877 | 878 | 879 |
| 880 | 881 | 882 | 883 | 884 | 885 | 886 | 887 | 888 | 889 | 890                                                     | 891 | 892 | 893 | 894 | 895 | 896 | 897 |
| 898 | 899 | 900 | 901 | 902 | 903 | 904 | 905 | 906 | 907 | 908                                                     | 909 | 910 | 911 | 912 | 913 | 914 | 915 |
| 916 | 917 | 918 | 919 | 920 | 921 | 922 | 923 | 924 | 925 | 926                                                     | 927 | 928 | 929 | 930 | 931 | 932 | 933 |
| 934 | 935 | 936 | 937 | 938 | 939 | 940 | 941 | 942 | 943 | 944                                                     | 945 | 946 | 947 | 948 | 949 | 950 | 951 |
| 952 | 953 | 954 | 955 | 956 | 957 | 958 | 959 | 960 | 961 | 962                                                     | 963 | 964 | 965 | 966 | 967 | 968 | 969 |
| 970 | 971 | 972 | 973 | 974 | 975 | 976 | 977 | 978 | 979 | 980                                                     | 981 | 982 | 983 | 984 | 985 | 986 | 987 |
| 988 | 989 | 990 | 991 | 992 | 993 | 994 | 995 | 996 | 997 | [ reached getoption("max.print") -- omitted 2 entries ] |     |     |     |     |     |     |     |

## Part g)

### Question-4:

Use the function paste to create the following character vectors of length 30:

(a) ("Label 1", "Label 2", ....., "Label 30")

\*Note that there is a single space between label and the number following.

(b) ("FN1", "FN2", ..., "FN30").

\*\*In this case, there is no space between fn and the number following.

#### Answer 4

##### Part a)

```
r<-paste("label",1:30,sep=" ")
```

r

```
69:1 (Top Level) R Script
Console ~/
> r<-paste("label",1:30,sep=" ")
> r
[1] "label 1" "label 2" "label 3" "label 4" "label 5" "label 6" "label 7" "label 8" "label 9" "label 10" "label 11" "label 12"
[13] "label 13" "label 14" "label 15" "label 16" "label 17" "label 18" "label 19" "label 20" "label 21" "label 22" "label 23" "label 24"
[25] "label 25" "label 26" "label 27" "label 28" "label 29" "label 30"
>
```

##### Part b)

```
p<-paste0("FN",1:30, sep="")
```

p

```
73:1 (Top Level) R Script
Console ~/
> f<- paste("FN", 1:30, sep = "")
> f
[1] "FN1" "FN2" "FN3" "FN4" "FN5" "FN6" "FN7" "FN8" "FN9" "FN10" "FN11" "FN12" "FN13" "FN14"
[15] "FN15" "FN16" "FN17" "FN18" "FN19" "FN20" "FN21" "FN22" "FN23" "FN24" "FN25" "FN26" "FN27" "FN28"
[29] "FN29" "FN30"
>
```

#### Question 5 :

Compound interest can be computed using the formula

$A = P \times (1 + R/100)^n$ , where P is the original money lent, A is what it amounts to in n years at R percent per year interest.

Write R code to calculate the amount of money owed after n years, where n changes from 1 to 15 in yearly increments, if the money lent originally is 10000 Rupees and the interest rate remains constant throughout the period at 11.5%.

#### Answer 5

#  $A = P \times (1 + R/100)^n$ , where P is the original money lent, A is what it amounts to in n years at R percent per year interest.

```
n<-c(1:15)
```

n

```
p=1000
```

p

```
m<-p*(1+11.5/100)^n
```

m

```
Console ~/
> n<-c(1:15)
> n
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
> p=1000
> m<-p*(1+11.5/100)^n
> m
[1] 1115.000 1243.225 1386.196 1545.608 1723.353 1921.539 2142.516 2388.905 2663.629 2969.947
[11] 3311.491 3692.312 4116.928 4590.375 5118.268
>
```

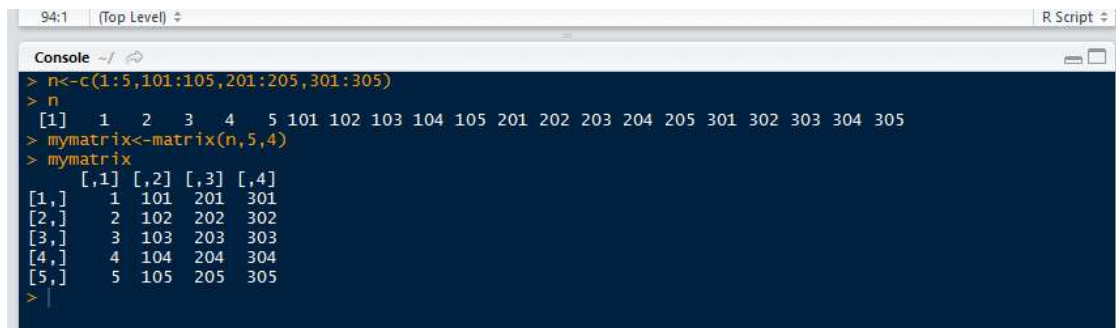
**Question 6** Generate the following matrices.

```
[,1] [,2] [,3] [,4]  
[1,] 1 101 201 301  
[2,] 2 102 202 302  
[3,] 3 103 203 303  
[4,] 4 104 204 304  
[5,] 5 105 205 305
```

**Answer 6**

**Rcommands:**

```
n<-c(1:5,101:105,201:205,301:305)  
n  
mymatrix<-matrix(n,5,4)  
mymatrix
```



The screenshot shows an R console window with the following text:

```
94:1 (Top Level) R Script  
Console  
> n<-c(1:5,101:105,201:205,301:305)  
> n  
[1] 1 2 3 4 5 101 102 103 104 105 201 202 203 204 205 301 302 303 304 305  
> mymatrix<-matrix(n,5,4)  
> mymatrix  
      [,1] [,2] [,3] [,4]  
[1,] 1 101 201 301  
[2,] 2 102 202 302  
[3,] 3 103 203 303  
[4,] 4 104 204 304  
[5,] 5 105 205 305  
>
```