|  |  |
| --- | --- |
| **EX03-03.cpp:** *Fall Through in switch case Statement* | |
| **Line#** | **Code** |
| 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 | #include <iostream>  #include <iomanip>  using namespace std;  enum DAY { SUN, MON, TUE, WED, THU, FRI, SAT };  long Max(long x, long y) {  if (x > y) return x;  else return y;  }  long Abs(long v) {  if (v < 0) v = -v;  return v;  }  const char\* WhatToEat1(DAY day) {  const char\* food = NULL;  if (day == MON) food = "Satay";  else {  if (day == WED) food = "Burger";  else {  if (day == FRI) food = "Pizza";  else {  if ((day==SUN)||(day==TUE)||(day==THU)||(day==SAT)) {  food = "Nasi Lemak";  }  else {  cout << "Invalid argument:" << day << endl;  }  }  }  }  return food;  }  const char\* WhatToEat2(DAY day) {  const char\* food = NULL;  switch (day) {  case MON:  food = "Satay";  break;  case WED:  food = "Burger";  break;  case FRI:  food = "Pizza";  break;  case SUN:  case TUE:  case THU:  case SAT:  food = "Nasi Lemak";  break;  default:  cout << "Invalid argument:" << day << endl;  }  return food;  }  int main() {  cout << "WhatToEat1\tWhatToEat2" << endl;  cout << "==========\t==========" << endl;  for (int d = SUN; d <= SAT; d++) {  cout << setfill(' ') << std::setw(10) << WhatToEat1((DAY)d)  << "\t" << WhatToEat2((DAY)d) << endl;  }  return 0;  } |