Cheatsheet - Reading the BMP180 Sensors on the qbcan

Pre-Setup

- Include the qbcan library (to use the BMP180 functions)
- Make a global variable of type BMP180

```
//Include the required libraries
#include <qbcan.h>

//Pressure sensor object
BMP180 bmp;
```

Setup

- Initialise the serial port with a baud rate of 9600
- Initialise the BMP180 chip by calling the begin() function
    - Wrap this within an error checking function to ensure that the chip is working

```
void setup()
{
  //Initialize serial connection for debugging
  Serial.begin(9600);
  Serial.println("Start!");

  // Initialize pressure sensor.
  if (bmp.begin())
    Serial.println("BMP180 init success");
  else
  {
    //In case of error let user know of the problem
    Serial.println("BMP180 init fail (disconnected?)\n\n");
    while(1); // Pause forever.
  }
}
```

n.b `while(1)` is a handy construct for an infinite loop (the code will never make it past this line). It works because 1 is seen as TRUE by the code, and with no instructions within the loop it just stays on this line.

Sensing loop

- First declare two local variables for storing the temperature and pressure.
- Then call the "getData" function of the bmp object, passing the temperature and pressure variables to be filled by the function.
- Finally a series of print() functions are used to display the temperature, pressure and some text.
- The delay() is used to limit the readings to every 500ms.

```
void loop()
{
  double T,P;
  // Get a new pressure reading:
  bmp.getData(T,P);

  //Display data
  Serial.print("Absolute pressure: ");
  Serial.print(P,2);
  Serial.println(" mb.");
  Serial.print("Temperature: ");
  Serial.print(T,2);
  Serial.println(" deg C.");

  delay(500);
}
```