STRING
- C-style string
- string class C++
- QString with Qt

---

"antonio "

| a | n | t | o | n | i | o | ␣ | \0 |

Ø

---
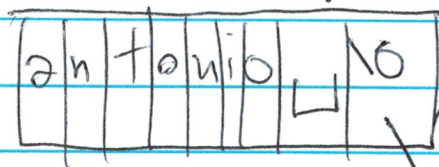
C-style char *s1 = "This ␣", *s2 = "is ␣ a ␣", *s3 = " string.";

char *concat = new char (strlen (s1) +
                                strlen (s2) +
                                strlen (s3) + 1);

strcpy (concat, s1);
strcat (concat, s2);
strcat (concat, s3);

. . .

delete concat;

```cpp
#include <string>
#include <iostream>

using namespace std;

...

string s1 ("This ");
       s2 ("is a "),
       s3 ("string.");

s1 += s2;
string s4 = s1 + s3;
```
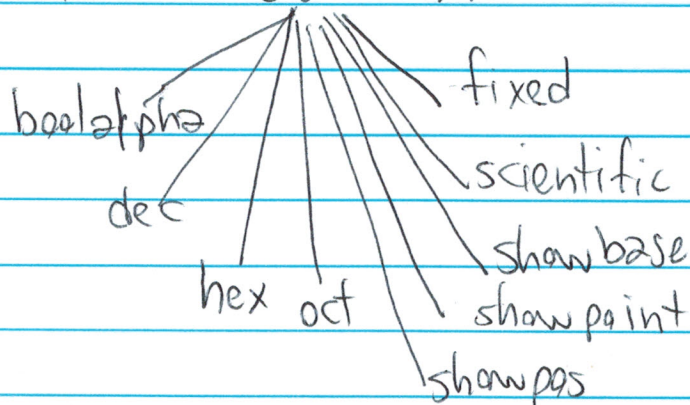
```cpp
getline (cin, s2);

cout << s2.length() << endl;        manipulators
cout << ... << flush;
```
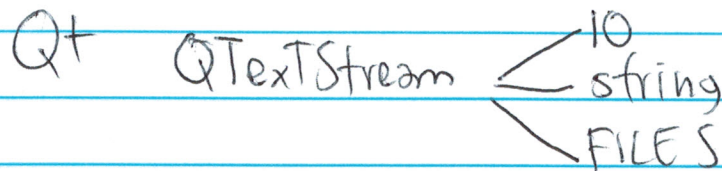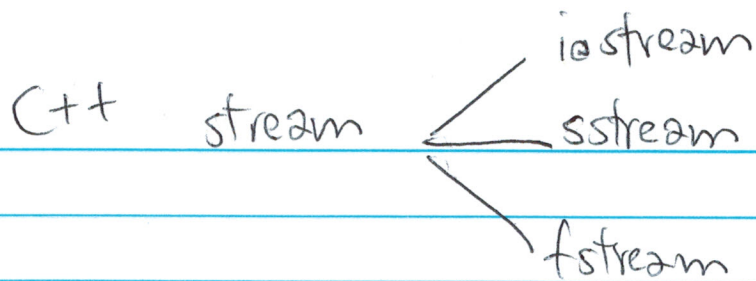
## 1.9    FORMAT FLAGS MANIPULATORS    (STREAM)

boolalpha
dec
hex   oct
fixed
scientific
showbase
showpoint
showpos

```cpp
cout << FLAG << double Value << endl;
```

C++    stream   ⟨ iostream
               sstream
               fstream

Qt    QTextStream  ⟨ IO
                   string
                   FILES

---

ex 1.11

        IO

```
int lucky = 7
float pi = 3.14;
double e = 2.71;


cout << lucky << '\t'
     << pi << '\t'
     << e << endl;
```

---

     STRING

```
ostringstream strbuf;

strbuf << lucky << '\t'
       << pi << '\t'
       << e << endl;
```
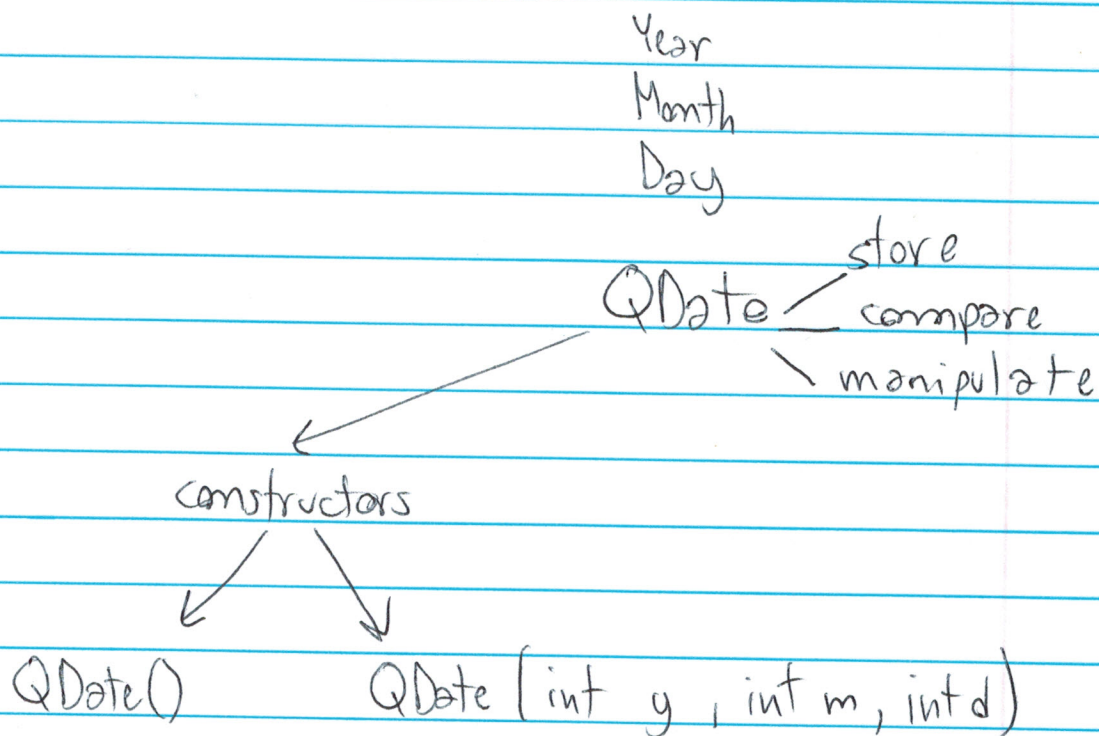
---

     FILE

```
ofstream outf;
outf.open ("filename");
outf << lucky << '\t';
     << pi       << '\t';
     << e << endl;
outf.close();
```

# 3.2.1 Stream and Date

Year
Month
Day

QDate ─── store
       ─── compare
       ─── manipulate

QDate ──→ constructors

constructors ──→ QDate()

constructors ──→ QDate (int y, int m, int d)

STATIC    QDate :: current Date()
          QDate :: from String (const QString & string,

                                    FORMAT);

          QtDateformat        QString

─────────────────────────────────────

          QDate :: add Days (qint64 ndays)
QDate ─── QDate :: add Months (int nm)
          QDate :: add Years (int ny)

# EXCEPTIONS  Ezust Ed1 CHAP21

```
try {
    ...    // code that generates EX
}
catch (ExceptionType & refName){

}
catch (                          ){

}
```

---

```
double  my function (char param) throw (int){

}
```

throws an
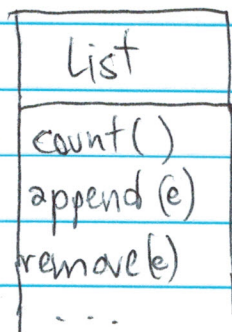exception

CONTAINERS
AGGREGATES
COLLECTIONS

↓ ARRAY

LIST    TREES    .HASH TABLES

---

| List |
| --- |
| count ( ) |
| append (e) |
| remove(e) |
| . . . |

int →  append (e:int)
       remove (e:int): int&

double →  append (e:double)
          remove (e:double):double&

myclass →  append (e: myclass&)
           remove (e: myclass&):double&

LIST → QList

QList < int >
QList < double >
QList < myclass >