کلمات کلیدی C++

کلمه کلیدی	شرح	مثال
and	مترادفی برای عملگر منطقی AND: &&	(x>0 and x<8)
and_eq	مترادفی برای عملگر تخصیصی AND بیتی: &=	b1 and_eq b2;
asm	اجازه می دهد تا اطلاعات مستقیما به اسمبلر فرستاده شود	asm ("check");
auto	کلاس ذخیره سازی برای اشیایی که فقط درون بلاک خودشان ایجاد می شوند	auto int n;
bitand	مترادفی برای عملگر AND بیتی: &	b0 = b1 bitand b2;
bitor	مترادفی برای عملگر OR بیتی: |	b0 = b1 bitor b2;
bool	معرف نوع بولین	bool flag;
break	یک حلقه یا عبارت switch را خاتمه می دهد	break;
case	در عبارت switch شرط کنترلی را مشخص می کند	case (n/10)
catch	فعالیت هایی را مشخص می کند که وقتی یک استثنا رخ می دهد باید اجرا شوند	catch(error)
char	معرف نوع صحیح	char c;
class	تعریف یک کلاس را مشخص می کند	class X { … };
compl	مترادفی برای عملگر منطقی NOT بیتی: ~	b0 = compl b1;
const	تعریف یک ثابت را مشخص می کند	const int s = 32;
const_cast	برای تغییر دادن اشیا از درون تابع عضو تغییرناپذیر استفاده می شود	pp = const_cast<T*>(p)
continue	در حلقه به ابتدای دور بعدی پرش می کند	continue;
default	حالت «وگر نه» در عبارت switch	default: sum = 0;

کلمه کلیدی	شرح	مثال
delete	حافظه ای را که با عبارت new تخصیص یافته آزاد می کند	delete a;
do	یک حلقه do..while را مشخص می کند	do {…} while …
double	نوع عددی حقیقی (ممیز شناور با دقت مضاعف)	double x;
dynamic_cast	برای اشاره گر داده شده، اشاره گر T* را بر می گرداند	pp = dynamic_cast<T*>p
else	بخش دیگر عبارت if را مشخص می کند	else n=0;
enum	برای تعریف نوع شمارشی استفاده می شود	enum bool {…};
explicit	باعث می شود تا از فراخوانی ضمنی یک سازنده جلوگیری شود	explicit X(int n);
export	دسترسی از واحد کامپایل دیگر را ممکن می سازد	export template<class T>
extern	کلاس ذخیره سازی برای اشیایی که بیرون از بلوک محلی تعریف شده اند	extern int max;
false	یکی از دو مقدار نوع bool	bool flag=false;
float	معرف نوع عددی حقیقی (ممیز شناور)	float x;
for	یک حلقه for ایجاد می کند	for (; ;) …
friend	در یک کلاس، تابع دوست ایجاد می کند	friend int f();
goto	باعث می شود تا اجرای برنامه به یک جملهٔ برچسب دار پرش کند	goto error;
if	یک عبارت شرطی if ایجاد می کند	if (n>0) …
inline	تابعی را تعریف می کند که متن آن تابع باید به جای فراخوانی آن درج شود.	inline int f();
int	معرف نوع عددی صحیح	int n;

کلمه کلیدی	شرح	مثال
long	برای تعریف انواع گسترش یافته صحیح و حقیقی استفاده می شود	long double x;
mutable	به توابع تغییر ناپذیر اجازه می دهد تا فیلد را تغییر دهند	mutable string ssn;
namespace	بلوک های فضای نام (میدان) را می شناساند	namespace best { int num; }
new	حافظه ای را تخصیص می دهد	int* p = new int;
not	مترادفی برای عملگر NOT منطقی : !	(not(x==0))
not_eq	مترادفی برای عملگر نابرابری : =!	(x not_eq 0)
opereator	برای تعریف سربارگذاری عملگرها به کار می رود	x operator++()
or	مترادفی برای عملگر OR منطقی : ||	(x>0 or x<8)
or_eq	مترادفی برای عملگر تخصیصی OR بیتی : =|	b1 or_eq b2;
private	اعضای خصوصی را در یک کلاس مشخص می کند	private: int n;
protected	اعضای حفاظت شده را در یک کلاس مشخص می کند	protected: int n;
public	اعضای عمومی را در یک کلاس مشخص می کند	public: int n;
register	مشخص کننده کلاس ذخیره سازی برای اشیایی که در ثبات ها ذخیره می شوند	register int i;
reinterpret_cast	یک شی با نوع و مقدار داده شده را بر می گرداند	pp = reinterpret_cast<T*>(p)
return	عبارتی که تابع را خاتمه می دهد و یک مقدار را بر می گرداند	return 0;
short	معرف نوع صحیح محدود	short n;
signed	برای تعریف انواع صحیح علامت دار به کار می رود	signed char c;

کلمه کلیدی	شرح	مثال
sizeof	تعداد بایت هایی که برای ذخیره سازی یک شی استفاده شده را بر می گرداند	n = sizeof(float);
static	کلاس ذخیره سازی برای اشیایی که در طول اجرای برنامه وجود دارند	static int n;
static_cast	برای اشاره گر داده شده یک اشاره گر T* بر می گرداند	pp = static_cast<T*>p
struct	تعریف یک ساختار را مشخص می کند	struct X {…};
switch	یک عبارت switch را مشخص می کند	switch (n) { … }
template	یک کلاس template را مشخص می کند	template <class t>
this	اشاره گری که به شیء جاری اشاره می کند	return *this;
throw	برای تولید یک استثنا به کار می رود	throw x();
true	یکی از مقادیر ممکن برای متغیرهای نوع bool	bool flag = true;
try	بلوکی را مشخص می کند که شامل مدیریت کننده استثنا است	try { … }
typedef	برای یک نوع موجود، مترادفی را تعریف می کند	typedef int Num;
typeid	شیئی را بر می گرداند که نوع یک عبارت را نشان می دهد	cout << typeid(x).name();
typename	مترادفی برای کلمه کلیدی class	typename X { … };
using	دستوری که اجازه می دهد تا پیشوند و فضای نام حذف شود	using namespace std;
union	ساختاری را مشخص می کند که اجزای آن از حافظه مشترک استفاده می کنند	union z { … };
unsigned	برای تعریف انواع صحیح بدون علامت استفاده می شود	unsigned int b;
virtual	تابع عضوی را تعریف می کند که در یک زیر کلاس نیز تعریف شده	virtual int f();

کلمه کلیدی	شرح	مثال
void	عدم وجود نوع بازگشتی را معین می کند	`void f();`
volatile	اشیایی را تعریف می کند که می توانند خارج از کنترل برنامه تغییر داده شوند	`int volatile n;`
wchar_t	نوع کاراکتری عریض (16 بیتی) برای unicode	`wchar_t province;`
while	یک حلقه while ایجاد می کند	`while (n > 0) …`
xor	معادلی برای عملگر OR انحصاری بیتی : ^	`b0 = b1 xor b2;`
xor_eq	معادلی برای عملگر تخصیصی OR انحصاری بیتی : =^	`b1 xor_eq b2;`