

Recent College Graduates

Do any of these statements describe your situation?

- ▶ Your résumé lists your college extracurricular activities.
- ▶ Your parent's address is listed as your permanent address.
- ▶ You list every job since you started working at 16.
- ▶ Your uncle is listed as a reference.

If you answered yes to even one of these, you're most likely a recent college grad, and the prospect of writing your first professional résumé may seem daunting. The biggest challenge is to fill the page so it doesn't only include your name, address, and college degree. (And increasing the font size to 16 points is not an option!)

As you sit down to draft your résumé, remember one thing: a lot of competition is out there for entry-level positions. If you are only 23 and just out of college, no employer expects you to have the track record of a 40-year-old. Don't worry about your lack of real-world experience. Instead, focus on the skills you've learned and the knowledge you've attained. Let's take a look at how to approach each section of your first résumé.

A special note for IT professionals-in-process: the final semester of your senior year is not the time to start on your résumé. Starting to build your resume as soon as you get into your heavy-duty IT courses is extremely important. Your projects are fresh in your mind then, and describing them will be easier when it's time to put them to good use. You never know when that great internship is going to come along—you want to be prepared!

Defining Your Career Objective

Forget the flowery language you find on most résumés. Simply put, recent college graduates need not include anything beyond the position they seek. Why?

- ▶ No one expects recent college graduates to know what they want in the long term from their career.
- ▶ Traditional career objective statements are usually ignored by employers scanning résumés in response to job ads.

No Plan Required

Employers know that most recent college graduates have spent the last two to five years earning their degree, taking general classes to provide the foundation for thinking logically and methodically (in theory, at least). They may be familiar with local college programs that offer more practical and hands-on programs, but they are primarily interested in finding candidates with working knowledge in IT systems and who appear trainable.

Your first postgraduation résumé is used to apply for your first "real" job in IT. College grads with a career plan are the rare exception. Truly, no one expects recent college graduates to know what their career will look like five to ten years out. Fortunately, a career is built one job at a

time, and many of the most successful and rewarding careers take unexpected detours that no one could predict at age 22. So, define the type of job you want as your first entry-level position and *describe your qualifications for that job*.

Objective Ignored

If you've spent hours perfecting the career objective portion of your résumé, you may be disappointed to learn that most employers won't take the time to admire your carefully crafted sentences. But you should understand why employers don't bother to read this statement.

- ▶ This statement is so generic, it could apply to any position in the company.
- ▶ The statement describes every other candidate applying for the position.
- ▶ Employers are reviewing résumés for specific job postings.

There's a double-edged sword, though. Even though your objective statement will probably be ignored, you still need to include it. Look at it this way: the hours you've invested won't go to waste. You just need to learn to maximize the effectiveness of your objective statement. And, the next section shows you how to do just that.

I'm Trainable!

Employers are looking for candidates who have a base set of IT skills and who are trainable. The top of your résumé is precious real estate for effectively addressing the question "Why should you hire me?" This is space that should be devoted to selling your credentials—both your educational and practical experience. Use this space to *imply a job objective, and advertise credentials and strengths as a potential employee*.

Check out the example shown here:

Qualifications

- ▶ *Self-motivated, customer-focused, recent college graduate with an exceptional ability to quickly master new software/hardware*
- ▶ *MCP, A+, and Network+ certified as part of earning undergraduate degree*
- ▶ *Six-month internship, supporting a multiple server Windows Server 2003 environment, servicing 200 users for an accounting firm*
- ▶ *Hired full-time postgraduation as the network administrator*
- ▶ *B.S. in Computer Science, Montclair State University, 2005*

Jennifer recently graduated from college and has been working for the past six months as a networking technician. She earned A+ and Network+ certifications as part of her undergraduate work and has gained valuable experience with on-the-job training through an internship program with a local accounting firm. Most important for Jennifer is to highlight her capacity to handle multiple important projects simultaneously (completing her coursework and holding down a full-time job) and her readiness to keep her skill set current with industry demands.

Defining Your Experience

Many recent IT college grads obsess over describing relevant, interesting, and impressive background experience. Some are so intimidated by their lack of hands-on experience, they resort to exaggerating or fabricating their personal histories.

Don't give in to this temptation. Stretching the truth on your résumé is always a mistake, particularly with IT positions. Most IT hiring managers include peer interviews as part of the interviewing process to help assess candidates' experience. A few simple questions by an experienced IT professional easily uncover how much you know. Exaggeration can mean automatic disqualification.

Relax. You have learned quite a bit in college. And, once you write this section of your résumé, you can tell your parents they got their money's worth. The following section can help you uncover your relevant experience. This involves two steps:

1. Uncovering your knowledge
2. Uncovering and applying your practical experience

What Have You Learned?

Ever wonder exactly what you learned from those endless theoretical college courses? A lot, in fact. Theory, design, and analysis courses go a long way on your résumé. First, these classes show you are trainable. You can learn. Employers like this. They also like knowing you know the basics—logical thinking, decision trees, redundancy planning, or whatever.

A simple way of discovering how much you know is to make a list of all the IT, math, and business courses you took in college, and then look up the descriptions of these courses in your school's course catalog as a refresher of what you've learned. Luckily, most course catalogs are available online these days, so it's not hard to copy-and-paste the descriptions into an impressive list. Table 6-1 is an example of the information you should look for and list.

TABLE 6-1 Skills Learned from College Courses

Course Title	Skills and Knowledge Gained
Introduction to Computer Science	Computer operating system (OS) architecture Hardware interfaces and OS software of computers Software engineering methodologies, including initial system specification, development, quality assurance, revisions, and deployment Develops initial design and programming skills using a high-level programming language (primarily C++/Java) Data structures, arrays, records, files, pointers, linked lists, trees, graphs, stacks, queues, and heaps
Computer Science II	Abstract behavior of classic data structures (stacks, queues, priority queues, tables, trees), alternative implementations, and analysis of time and space efficiency Recursion Object-oriented and functional programming Models of computation

TABLE 6-1 Skills Learned from College Courses (continued)

Course Title	Skills and Knowledge Gained
Introduction to Software Engineering	Introduction to the concepts, methods, and current practice of software engineering The study of large-scale software production, software life-cycle models as an organizing structure, principles and techniques appropriate for each stage of production
UNIX Operating System	UNIX system administration, including tools and utilities, shells, and mail and news administration
C Programming	Structured techniques, pointers, structures, classes, declarations, tools and libraries, I/O and file manipulation, application compilation, and abstract programmer-defined objects
Java Programming	Java object-oriented programming (OOP) principles, graphical user interface programming, SQL databases, and client/server programming techniques
Operating Systems and Computer Architecture	Systems structure and systems evaluation Memory management and process management
Programming Fundamentals	C Java C++
Relational Databases	Relational algebra, views, queries, normal forms, optimization, and incrementality Other models for databases: hierarchical and network models The entity-relationship model, knowledge bases, and exceptions Distributed databases Applications Programming in SQL and Oracle
Software Engineering and Object-Oriented Development	Large-scale application design using entity-relation and object-oriented models Management of large-scale projects, including version control, document traceability, and distributed development Testing, validation, and verification Introduction to formal methods, simulation as a tool Large-scale, team-oriented project
Systems Analysis/Design Theory	Fundamentals of development of successful computer-based information systems, with an emphasis on the roles of systems analysts, programmers, users, and management Definition of user requirements Systems analysis life cycle
Database Theory	In-depth coverage of the content of database management systems (DBMS), including physical and logical database structures using Oracle
Introduction to SQL	Foundations of SQL commands for business applications
HTML Programming	Web design, HTML, FrontPage, Dreamweaver, JavaScript

TABLE 6-1 Skills Learned from College Courses (*continued*)

Course Title	Skills and Knowledge Gained
AS/400 Programming	Control Language (CL) for message handling, data queues, data areas, logic statements, program calls, and asynchronous jobs
Oracle	Database design, objects, data integrity, security, and performance tuning
Networking Essentials	Networking architecture, transmission concepts, and management OSI model TCP/IP, server installation, applications, user management, workgroup management, printing, and security Windows Server Client/server Network cabling
VB Programming	VB, database creation and access, inter-application communication, advanced printing techniques, and graphics Event-driven programs Advanced access programming Graphical user interface (GUI) design
C++ Programming	Object-oriented programming (OOP) Microsoft Foundation Classes GUI programs Pointers, memory management, and user-defined types Object linking and embedding (OLE) automation Dynamic Data Exchange (DDE) Large application design and development

Uncovering Your Practical Experience

Most computer science or MIS programs include hands-on projects. Some come in the form of course projects, others in senior projects and, best of all, internships.

Course projects and senior projects are practical applications of your theoretical knowledge. With the fast-paced IT environment out there, finding a computer science program that teaches the latest and greatest of the commercial applications and languages used in today’s business environment is difficult. And, it is next to impossible to hire leading-edge developers to teach in a college environment or to find college professors who are truly involved in real IT business projects. Nonetheless, your college projects offer considerable experience that can be represented as practical experience on your résumé.

Make a list of the experience you accomplished. The things to pull out of this list include:

- ▶ Real-world application languages you can program in (C#, .NET, Java, HTML, XML)
- ▶ Use of commercial applications (Microsoft Excel, Microsoft Word, Macromedia Flash)
- ▶ Use of real-world processes for planning, design, and project management (Microsoft Visio, Microsoft Project, Microsoft Visual Studio)
- ▶ Management of networking systems (Windows Server 2003, Novell NetWare, Sun Solaris)
- ▶ Operation of computer operating systems (Windows XP, Windows 2000 Professional, UNIX, Linux)
- ▶ Development of databases using commercial software (Microsoft SQL, Oracle, Sybase)
- ▶ Understanding of ERP systems (SAP)

Internships are more common with IS degrees from business management schools. They offer an excellent opportunity to gain hands-on experience in a true business environment. Most last only a single semester, so you can't expect to get much real experience, but you do get a sense of what it means to work within a team of IS people.

If you've completed an internship, it shows you have initiative. Hiring managers value that more than the actual experience you may have gained. The key things employers look for from internships are:

- ▶ Application of basic skill set
- ▶ Introduction to real-world applications, programming languages, and development processes
- ▶ Accountability
- ▶ Ability to perform under a deadline
- ▶ Project planning and documentation experience
- ▶ Project ownership and responsibility
- ▶ Ability to work as part of a team

Pulling It All Together

The best way to describe the skills you pick up from course projects and internships is to use a skills-based résumé or functional résumé layout. A *skills-based résumé* works best when your skills are more impressive than your job titles. As the name suggests, this kind of résumé highlights skills, while deemphasizing actual employment dates by presenting a condensed work history at the bottom of the résumé. This is a great way to present significant skills gained from a position that lasted a short period of time—such as an internship.

A skills-based résumé organizes your skills into categories and lets you list your knowledge under these headings, such as the following:

- ▶ Programming and Analysis
- ▶ Database Design
- ▶ Computer System Administration
- ▶ PC Support and User Training
- ▶ Network Analysis and Documentation
- ▶ Customer Support
- ▶ Computer Math
- ▶ Data Integrity
- ▶ Computer Applications
- ▶ Systems Analysis
- ▶ Internet Development
- ▶ Business Management
- ▶ Operations Management

The next step is to review the job posting you're applying for. Identify the key skills they're seeking and pick out the relevant skill headings you should include. Any other knowledge and skills you feel would make you a better candidate for the position should be categorized under a functional heading of "Additional Skills." Practical experience should be interspersed with the knowledge and skills you have acquired. Look to include three to six supporting statements under each skills heading, including examples of your practical experience.

Figure 6-1 shows a sample résumé that doesn't maximize college experience well. Figure 6-2 shows the same information presented differently, to create a much more effective résumé. Without much editing, we were able to draw attention from the fact that Nick has only been working as a helpdesk technician for three months, has yet to earn his B.A. degree, and had previous unrelated experience. Here's what we did:

- ▶ Removed the objective statement. It had too many buzzwords and was too generic.
- ▶ Introduced the "Profile" section.
- ▶ Moved up and renamed the "Special Skills" section to "Technical Skills," and then listed skills in bulleted format.
- ▶ Added computer coursework information to provide more details around your major course of study.
- ▶ Eliminated "Interests" section, which is irrelevant to employers.