



- A** Absentees
- Absenteeism table, 6, 7
 - Calendar table, 8
 - discharging personnel, 4
 - long-term illnesses, 7
 - puzzle, 4–8
 - table, 4
- ABS() function, 103, 104, 237
- AGE() function, 7
- Age ranges for products puzzle, 261–62
- Aggregate functions, on empty sets, 31
- Airlines and pilots
- exact relational division, 90
 - pilots table, 88
 - planes table, 88
 - puzzle, 88–91
 - relational division, 89
- ALL() predicate, 255
- Alpha data puzzle, 19–20
- ANDs, nested, 265
- Anesthesia puzzle, 9–15
- Anesthesiologist procedures
- concurrent, 12–13
 - elimination, 11
 - overlapping, 10
 - payment, 9
- Armes, Jim, 146
- Attenborough, Mary, 197
- Available seats puzzle, 34–36
- Average
- moving, 152–54
 - rule, 174
 - sales wait puzzle, 126–28
- AVG() function, 123, 127, 174
- B** Backward-looking sums, 12
- Badour, Bob, 209, 257
- Barcodes
- pseudofunction, 238–39

- puzzle, 237–41
 - set-oriented, declarative answer, 240–41
 - Becher, Johannes, 219
 - BETWEEN predicates, 3
 - in CHECK() clause, 19
 - for durations, 39
 - for reading/maintaining code, 93
 - subqueries in, 187
 - temporal version, 22
 - Bin-packing problem, 288
 - Block of seats puzzle, 190–91
 - Blumenthal, Nigel, 148–51
 - Bose-Nelson solution, 242
 - Boxes
 - intersecting pairs, 257
 - n-dimensional, 257
 - puzzle, 257–60
 - Bragg, Tom, 112
 - Brouard, Frédéric, 106
 - Brown, Robert, 215
 - Buckley, Brian K., 141
 - Budgeted table, 210
 - Budgeting puzzle, 169–71
 - Budget versus actual puzzle, 208–11
 - Buying all products
 - deeply nested query, 130
 - puzzle, 129–31
 - tables, 129
- C**
- Cady, C. Conrad, 208
 - Calculations
 - puzzle, 297–99
 - query text, 297
 - Calendar table, 8
 - Campbell, Brendan, 53
 - Cartesian distance formula, 309
 - CASE expressions, 32, 46
 - collapsing SELECT statements into, 54
 - ELSE NULL and, 102
 - GROUP BY clause and, 150–51
 - in LIFO-FIFO inventory puzzle, 285–87
 - optimization tricks, 46–47
 - self-joins and, 287
 - as UNIONs replacement, 110, 184
 - WHEN clauses, 47
 - WHERE clause, 219
 - CAST expression, 64
 - Catching the next bus
 - bus schedule, 282
 - puzzle, 280–82
 - table, 280
 - without comparisons, 281–82
 - Categories table, 176
 - CEILING() function, 193
 - Chacha, Yogesh, 29
 - Chains, 28
 - Characteristic functions, 46
 - CHECK() clause, 16–17, 28, 190
 - BETWEEN predicates in, 19
 - complex SQL in, 21
 - constraints, 19, 47, 182
 - subqueries and, 22, 191
 - substrings in, 19
 - Chupella, Jim, 4
 - Claims status puzzle, 48–52
 - defendant, 49
 - numeric claim sequence, 50
 - Clock table, 15
 - COALESCE() function, 32, 57, 65, 106, 136, 156

- to current timestamp, 300
 - parameter list inspection, 174
 - Collapsing table by columns puzzle, 215–17
 - Columns
 - alpha data, 19–20
 - collapsing tables by, 215–17
 - in GROUP BY clause, 100
 - in SELECT list, 100
 - Common table expression (CTE), 41, 63
 - gaps, 233
 - LIFO-FIFO inventory, 290
 - recursive, 233
 - sales promotion, 189
 - Comparison operators, 47
 - Comparison predicates, 105
 - Computing depreciation puzzle, 137–40
 - Computing taxes
 - hierarchy, 132
 - puzzle, 132–36
 - table, 132, 133
 - taxing authorities, 134, 136
 - See also* Taxes
 - Constraints
 - CHECK(), 19, 47, 182
 - FOREIGN KEY, 56
 - No_Overlaps, 191
 - PRIMARY KEY, 191
 - string, 222
 - testing, 165
 - UNIQUE, 2
 - Consultant billing puzzle, 141–44
 - Contained in or equal to, 115
 - Contiguous groupings puzzle, 254–56
 - ConwayMike, 64
 - COUNT(*), 90
 - in average sales wait puzzle, 128
 - in budgeting puzzle, 171
 - in personnel problem puzzle, 213
 - testing against, 162
 - COUNT (DISTINCT <expression>)
 - aggregate function, 203, 213
 - Counting fish puzzle, 172–75
 - Covering index, 296
 - CREATE TABLE statement, 1
 - CROSS JOINS, 89, 298
 - in getting all possible pairs, 163
 - as multiplication operator, 89
 - Curly brackets, 301
- D** Data
- alpha, 19–20
 - false, 64
 - Database Programming & Design*, xi, 115
 - Dataflow diagrams (DFDs)
 - bubbles, 112
 - diagram name, 112
 - flow lines, 112
 - puzzle, 112–14
 - table, 112
 - Data scrubbing
 - problem, 304
 - proprietary looping cursor, 304
 - puzzle, 304–5
 - system utility function, 305
 - Date, Chris, 64, 65, 89, 115
 - Dautbegovic, Dzavid, 40, 249
 - Davison, Allen, 306
 - DAYS() function, 127
 - DB2 On-Line Magazine*, 91

- DBMS magazine, xi
 - DECODE() function, 47
 - DELETE statement, 5
 - DeMorgan's law, 96
 - DENSE_RANK() function, 85
 - DENSE_RANK() OVER (<window expression>) function, 224
 - Depreciation
 - computing, 137–40
 - cost table, 137
 - lifespan, 137
 - manufacturing hours table, 137
 - De Rham, Abbott, 179
 - Derived tables
 - avoiding, 307
 - puzzle, 306–8
 - Desai, Bipin C., 115
 - DISTINCTs, 114, 131
 - Double duty puzzle, 148–51
 - Duplicates
 - dropping incorrectly, 222
 - potential, 218–20
 - row, 179–80
 - Dwyer, Trevor, 105
- E**
- Elements table, 164
 - Employees
 - billings, 143
 - candidate skills, 75
 - effective date, 143
 - firing rules, 4–5
 - mechanic, 71
 - numbering rows within, 67
 - salary changes, 61–62
 - severity points, 4
 - total charges, 141–42
 - total earnings, 37
 - workload, 37
 - Employment agency
 - candidates, 75, 78
 - DOT, 76
 - job orders, 76
 - puzzle, 75–79
 - Empty sets
 - aggregate functions on, 31
 - returning, 120
 - Ersoy, Cenk, 45
 - Esperant, 129
 - EXACT function, 1
 - EXCEPT ALL operator, 229
 - EXCEPT operator, 118, 258
 - EXISTS() predicate, 90, 130
 - for checking “blocking pairs,” 270
 - nested, 91
 - Extrema functions, 53
- F**
- Federl, Carl C., 96
 - FIFO (first in, first out). *See* LIFO-FIFO inventory
 - Finding a pub
 - Cartesian distance formula, 309
 - puzzle, 309–11
 - square neighborhood, 310
 - Finding equal sets puzzle, 115–20
 - Finding mode computation puzzle, 123–25
 - Find last two salaries puzzle, 60–68
 - First Normal Form (1NF), 55, 104
 - Fiscal year tables puzzle, 1–3
 - Flags, plus/minus representation, 34
 - Flancman, Alan, 103
 - FLOOR() function, 193

FOREIGN KEY constraint, 56
Friends of Pepperoni, 183
FROM clauses, 58, 231
Frontera, Mark, 169
FULL OUTER JOINs, 57, 115, 308
Fundamentals of Database Systems, 115

G Gallagher, Karen, 92
Gammans, Scott, 21
Ganji, Kishore, 58
Gaps
 CTE, 233
 no, 228
 puzzle (version one), 227–29
 puzzle (version two), 230–333
 starting/ending values, 227
Gilson, John, 215
Gora, Mike, 158
Graduation
 Categories table, 176
 puzzle, 176–78
Greedy algorithms, 288
GROUP BY clause, 59, 79, 99, 203, 206
 in aggregates creation, 100
 CASE expression and, 150–51
 columns, 100
 inside correlated subqueries, 98
 sort invocation, 146–47
Grouped tables, 79
Groups
 contiguous, 254–56
 empty, 170
 non-NULL values in, 110
Gusfield, Dan, 278

H Halloran, Donald, 254
Harakiri, Mikito, 257, 258
Harvey, Roy, 86, 89, 148
HAVING clauses, 38, 83, 128
 extending, 161–62
 predicates, reducing, 188
Hiner, Ron, 224
Hotel reservations puzzle, 21–23
Hotel room numbers
 puzzle, 224–26
 table, 224
 WATCOM approach, 224, 225
 working table data, 224
Hughes, Bert C., 14
Hughes, Dave, 43

I Indexes, covering, 296
Indexing, 97
IN predicate, 164
 expansion into equality predicate,
 105
 as test for membership, 115
INSERT INTO statement, 23
INSERT statements, 22
Insurance losses
 correct policy tables, 158–59
 customer table, 158
 losses table, 159
 puzzle, 158–62
Intelligent Enterprise, xi
International Standard Book Number
 (ISBN), 203
Interpolation, 122
Interpolation, linear, 122

- An Introduction to Data Base Systems*,
115
- Introduction to Database Systems*, 90
- Inventory adjustments puzzle, 145–47
- Irving, Robert W., 278
- ISNUMERIC() function, 238
- ISO naming conventions, 301
- Israel, Elizabeth, 261
- J**
- Jaiswal, Anilbabu, 174
- Jilovec, Gerhard F., 137
- Joe Celko's Trees and Hierarchies in SQL
for Smarties*, 136
- JOINS
subqueries in, 210
See also specific types of JOINS
- Journal table, 157
- Journal updating puzzle, 155–57
- Julian workdays, 8
- Junk mail puzzle, 80–81
- K**
- Kass, Steve, 255
- Keeping a portfolio puzzle, 24–28
- Knuth, Donald E., 279
- Kubu, Sissy, 192
- Kuznetsov, Alexander, 101, 119, 131
- L**
- Landlord puzzle, 92–93
- Larsen, Sheryl, 91
- Lawrence, Peter, 196, 197
- LEFT OUTER JOINS, 70, 92, 171,
210, 248
joining tables with, 306
- on two columns, 175
- Legal events, ordering, 49
- LIFO-FIFO inventory
bin-packing problem, 288
bins in table, 289
CTE, 290
derived table and CASE
expression, 285–87
puzzle, 283–91
table, 283
UPDATE statements, 287
- LIFO (last in, first out). *See* LIFO-FIFO
inventory
- LIKE clause, 276
- LIKE predicate, 20
- Linear interpolation, 122
- M**
- Magazine
distribution database table, 94
newsstand selection, 94–95
puzzle, 94–103
- Manko, Gerard, 69
- Manufacturing cost, 137, 138
- MAX() function, 17, 31–32, 38, 43, 54,
59, 124
highest non-NULL value, 110
as safety check, 171
values, 210–11
values, optimizer finding, 120
- McDonald, J. D., 188
- McGregor, Keith, 94
- Mechanics puzzle, 69–74
- Medal, Leonard C., 9, 48, 149
- Melissa Data, 220
- Mello, Vinicius, 197
- Merging time periods puzzle, 34–36

- Milestone
 - puzzle, 107–11
 - self-joins, 108
 - service delivery, 107
 - subquery expressions, 108–9
 - table structure, 107
 - UNION ALL operators, 109–10
 - MIN() function, 17, 54, 245
 - values, 210–11
 - values, optimizer finding, 120
 - MINUS operator, 118
 - Missing values, 54
 - Mode computation, finding, 123–25
 - Mode() function, 125
 - MOD() function, 245–46
 - modulus, changing, 253
 - as vendor extension, 246
 - Moreau, Tom, 231
 - Moreno, Francisco, 51, 59, 81, 100, 118, 139, 210
 - Moving average
 - holding, 152
 - predicate construction, 153
 - puzzle, 152–54
 - Multiple-column row expressions, 98
- N**
- Nebres, Diosdado, 135
 - Nested function calls, 239
 - Nested ORs, 265
 - Nested sets, 136
 - Nested subqueries, 258
 - Nguyen, Linh, 143
 - Noeth, Dieter, 67, 68, 232
 - NOT condition, 11
 - NOT EXISTS predicate, 130, 205
 - maximizing performance, 276
 - set difference replacement, 118
 - traditional test, 117
 - NOT NULL, 1, 47
 - NULLs, 5, 87
 - handling, 124
 - for missing values, 54
 - multiple, 30
 - return of, 174
 - Numbering functions, 224
 - NUMBERS(*) function, 224
- O**
- Ocelot software, 19
 - Odegov, Andrey, 44, 65
 - OLAP/CTE, 68
 - OLAP functions, 43
 - running totals, 147
 - SQL-99, 153
 - support, 85
 - Omnibuzz, 232, 233
 - One in ten
 - puzzle, 103–6
 - table, 103
 - ORDER BY clause, 85, 226
 - ORs, nested, 265
 - Orthogonality, 123, 124
 - OUTER JOINS, 56–57, 63
 - Gupta-style extended equality, 58
 - persistent/transient in, 92
 - with RANK() function, 66
 - in Select clauses, 145
 - self, 245
 - OVERLAPS predicate, 22
 - OVER() window clause, 147

P

Padding, 239
Pairs of styles puzzle, 179–82
Paradox table, 69
Pascal, Fabian, 60, 64
Pepperoni pizza puzzle, 183–85
Permutations
 defined, 163
 factorial number of, 163
 puzzle, 163–68
Personnel problem puzzle, 212–14
Personnel Schedule table, 302
Petersen, Raymond, 126
Playing the ponies
 horse name table, 223
 puzzle, 221–23
 table, 221
Pointer chains, 25
Poole, David, 261
POSITION function, 151
Potential duplicates
 defined, 218
 expression arrangement, 219
 mailing list cleanup packages, 220
 puzzle, 218–20
Predicates
 ALL, 255
 BETWEEN, 3, 19, 22, 39, 93
 comparison, 105
 EXISTS, 90, 91, 130
 HAVING clause, 188–89
 IN, 105, 115, 164
 LIKE, 20
 NOT EXISTS, 117, 118, 130
 OVERLAPS, 22
 SIMILAR TO, 239
PRIMARY KEY constraint, 191
Primary keys

 covering index, 296
 multiple columns, 175

Printers

 common, 30, 32
 load balancing, 30
 LPT numbers, 31
 scheduling, 29–33
 unassigned, 32

Puzzles. *See* SQL puzzles

R

Race, Daren, 212
RANK() function
 defined, 85
 hidden sort, 67
 OUTER JOINS with, 66
Raval, Nayan, 248
REFERENCING clause, 72
Referential integrity, 70
Regular expression predicate, 20
Relational division, 89
 COUNT(*) version, 90
 exact, 90
REPLACE() function, 243
REPLICATE() function, 243
Report formatting
 experimental table, 246
 puzzle, 244–53
 two-across solution, 245
Reservations
 puzzle, 190–91
 rule, 190
 table, 190
Rightsizing, 16
Robertson, Gillian, 130
Romley, Richard, 13, 31, 36, 39, 54,
 63, 98, 132, 176, 188, 269

- ROW_NUMBER() function, 43, 85
 - Rows
 - adding/deleting, 294
 - “blocking pairs,” 270
 - duplicate, 179–80
 - harvesting, 194
 - inserted, 5
 - Running totals, 147
 - Russian peasant’s algorithm, 192, 197, 199
- S**
- Sales promotion
 - clerk performance, 186
 - CTE, 189
 - puzzle, 186–89
 - Samet, Alan, 41
 - Scalar subqueries, 121, 170
 - inside function calls, 156
 - results, 171
 - See also* Subqueries
 - Scalzo, Bert, 6
 - Scheduling printers puzzle, 29–33
 - Scheduling service calls
 - double booking, 300
 - Personnel Schedule table, 302
 - process, 300
 - puzzle, 300–303
 - Security badges puzzle, 16–18
 - Sedgewick, Robert, 168
 - Select list
 - columns, 100
 - improper creation, 181
 - subqueries in, 145
 - SELECT statement, 26–27, 173
 - collapsing, 54
 - as grouped query, 63
 - independent scalar, 71
 - innermost, 124
 - OUTER JOIN queries in, 145
 - outermost, 124
 - scalar subquery, 50–51
 - Self-joins, 39, 66, 108, 180
 - CASE expression and, 287–88
 - milestone puzzle, 108
 - UNION versus, 149–50
 - Sequence Auxiliary table, 196, 228
 - Sequences
 - gaps, finding (version one), 227–29
 - gaps, finding (version two), 230–33
 - numbering, resetting, 18
 - Sequence table, 262
 - Service delivery, 107
 - Set difference, 116
 - Set operations, 100
 - Sets
 - empty, 31, 120
 - equal, finding, 115–20
 - nested, 136
 - Shankar, Mr., 86
 - Sharma, R., 307
 - Sherman, Phil, 40
 - Shirbu, Sorin, 51
 - SIGN() function, 294
 - ABS() function combination, 103, 104
 - return, 103
 - SIMILAR TO predicate, 239
 - Sine function calculation puzzle, 121–22
 - Sizintsev, Dmitry, 249
 - Sudoku

- defined, 263
- delete statements, 264
- known cells table, 265–66
- puzzle, 263–66
- Sorting strings
 - Bose-Nelson solution, 242
 - Fike's algorithm, 242
 - puzzle, 242–43
 - See also* Strings
- SQL-89, tabular query expressions, 181
- SQL-92
 - CHECK() clause in subqueries and, 191
 - orthogonality, 123, 124
 - row/table constructors, 105
 - Select list subqueries, 145
 - set operators, 100
- SQL-99, OLAP functions, 153
- SQL-2003, OLAP functions, 147
- SQL for Smarties*, 242
- SQL puzzles
 - absentees, 4–8
 - age ranges for products, 261–62
 - airlines and pilots, 88–91
 - alpha data, 19–20
 - anesthesia, 9–15
 - available seats, 34–36
 - average sales wait, 126–28
 - barcodes, 237–41
 - block of seats, 190–91
 - boxes, 257–60
 - budgeting, 169–71
 - budget versus actual, 208–11
 - buying all products, 129–31
 - calculations, 297–99
 - catching the next bus, 280–82
 - claims status, 48–52
 - collapsing table by columns, 215–17
 - collection, xii
 - computing depreciation, 137–40
 - computing taxes, 132–36
 - consultant billing, 141–44
 - contiguous groupings, 254–56
 - counting fish, 172–75
 - dataflow diagrams, 112–14
 - data scrubbing, 304–5
 - defined, xi
 - derived tables, 306–8
 - double duty, 148–51
 - employment agency, 75–79
 - finding a pub, 309–11
 - finding equal sets, 115–20
 - finding mode computation, 123–25
 - find last two salaries, 60–68
 - fiscal year tables, 1–3
 - graduation, 176–78
 - hotel reservations, 21–23
 - hotel room numbers, 224–26
 - insurance losses, 158–62
 - inventory adjustments, 145–47
 - journal updating, 155–57
 - junk mail, 80–81
 - keeping a portfolio, 24–28
 - landlord, 92–93
 - LIFO-FIFO inventory, 283–91
 - magazine, 94–102
 - mechanics, 69–74
 - merging time periods, 34–36
 - milestone, 107–11
 - moving average, 152–54
 - one in ten, 103–6

- pairs of styles, 179–82
 - pepperoni pizza, 183–85
 - permutations, 163–68
 - personnel problem, 212–14
 - playing the ponies, 221–23
 - potential duplicates, 218–20
 - report formatting, 244–53
 - sales promotions, 186–89
 - scheduling printers, 29–33
 - scheduling service calls, 300–303
 - security badges, 16–18
 - sine function calculation, 121–22
 - Sudoku, 263–66
 - sorting strings, 242–43
 - stable marriages problem, 267–79
 - stock trends, 292–96
 - teachers, 53–55
 - telephone, 56–59
 - test results, 86–87
 - top salespeople, 82–85
 - two of three, 203–7
 - ungrouping, 192–99
 - wages of sin, 37–44
 - widget count, 200–202
 - work orders, 45–47
- Stable marriages problem
- backtracking algorithms, 267, 269
 - defined, 267
 - goal, 267
 - happy versus stable marriages, 267
 - n=8 code, 271–75
 - puzzle, 267–79
 - query, 270
 - solutions, 267
 - Unstable table, 276
- Stearns, Bob, 190
- Steffensen, J. F., 122
- Stock trends
- puzzle, 292–96
 - triggers, 293
 - VIEWS, 292
- Stock value calculation, 283–84
- Stored procedures, 294, 295
- Strings
- characters, converting, 167
 - constraints, 222
 - inserting, 167
 - oversized, 239
 - padding, 239
 - sorting, 242–43
- STUFF function, 167
- Styles
- pairs of, 179–82
 - table, 179
- Subqueries
- in BETWEEN predicate, 187
 - CHECK() clause and, 22, 191
 - converting to VIEWS, 63
 - correlated, 98
 - in JOINS, 210
 - nested, 258
 - scalar, 121, 170
 - in Select list, 145
- Subsets, 115
- SUBSTRING() function, 151
- Substrings, in CHECK() clause, 19
- SUM() function, 46, 107, 143, 306
- T** Tables
- Absenteeism, 6, 7
 - absentees, 4
 - Budgeted, 210
 - buying all products, 129

- Calendar, 8
 - Categories, 176
 - Clock, 15
 - collapsing by columns, 215–17
 - computing taxes, 132
 - deconsolidating, 192
 - derived, 306–8
 - DFD, 112
 - dividend, 91
 - Elements, 164
 - emptying, 195
 - fiscal year, 1–3
 - fish data, 172
 - grouped, 79
 - insurance losses, 158–59
 - Journal, 157
 - junk mail, 80
 - magazine distribution database, 94
 - next bus, 280
 - one in ten, 103
 - Paradox, 69
 - Personnel Schedule, 302
 - pilots, 88
 - planes, 88
 - playing the ponies, 221
 - primary keys, 175
 - redesign, 22
 - reservations, 190
 - Sequence, 262
 - Sequence Auxiliary, 196, 228
 - tax computation, 133
 - Team, 71, 72
 - temporary, 96, 148
 - time slots, 153–54
 - Unstable, 276
 - Table scans, 157, 198
 - Tabular query expressions, 181
 - Taxes
 - computing, 132–36
 - current rates, 136
 - multiple authorities, 132
 - table, 133
 - Team table, 71, 72
 - Temporal functions, 157
 - Temporary tables, 96, 148
 - Teradata, 68
 - Test results puzzle, 86–87
 - Teter, Chris, 304
 - Thompson, Adam, 97
 - Tilson, Steve, 24
 - Timestamps, 153, 300
 - Top salespeople puzzle, 82–85
 - Triggers
 - bulk insertions and, 293
 - on insertion, 182, 241
 - proprietary language, 293
 - writing, 293
 - Two of three puzzle, 203–7
 - Tymowski, Luke, 37
- U** Ungrouping
- answer table, 193
 - approach comparison, 199
 - defined, 192
 - JOIN to a table, 198
 - puzzle, 192–99
 - Russian peasant's algorithm, 192, 197, 199
 - Sequence Auxiliary table, 196
 - table emptying, 195
 - table scan, 198
 - working tables, 194–95
 - UNION ALL operators, 63, 109–10

- in FROM clause, 231
 - in milestone puzzle, 109–10
- UNION operators, 11, 53, 62, 297
- CASE expression replacement, 110, 184
 - self-joins versus, 149–50
- UNIQUE constraints, 2
- Unstable table, 276
- UPDATE statement, 6–7, 153, 295
- V**
- VALUES() expression, 240
- Van de Pol, Lex, 14
- VIEWS
- aggregate information, 221
 - avoiding with subquery table expression, 63
 - combining, 36, 57
 - converting subqueries to, 63
 - CTE expression, 96
 - with joined aggregate, 96
 - materializing, 58
 - outer-joined, 58
 - portability and, 129–30
 - stock trends puzzle, 292
 - summarizing from, 142
 - WITH CHECK OPTION, 22
- W**
- Wade, Larry, 75
- Wages of sin puzzle, 37–44
- WATCOM SQL, 135, 224, 225
- Weisz, Ronny, 218
- Wells, Jack, 60
- WHEN clauses, 47
- WHERE clauses, 11, 31, 219, 245
- WHILE loops, 242
- Widget count puzzle, 200–202
- Wiitala, Mark, 151
- Wilton, Tony, 242
- WINDOW clause, 68
- Wirth, Niklaus, 279
- Working days, 8
- Work orders puzzle, 45–47
- Y**
- Young, Brian, 107
- Young, Ian, 165