Index

enum class (cont.)
- use cases (cont.)
  - namespace pollution, avoiding, 339–340
  - overloading disambiguation, 340–343
- weakly typed C++03 enumerators, drawbacks to, 333–335

enumerations. *See also* opaque enumerations
- comparisons in C++20, 335n1
- underlying types (UTs)
  - description of, 829–830
  - further reading for, 834
  - potential pitfalls, 832–833
  - use cases, 830–832

enumerators, as compile-time constants, 164
errors
- compiler warnings as, 150
- compile-time, 22

escalation, 374
essential behavior, 102
event-driven callbacks, 603–604
exception agnostic, 644, 1126
exception-free path, 1134, 1136, 1143
exception safe, 644, 1126
exception specifications, 593. *See also* **noexcept** exception specifications
- conditional, 1091–1092
- constraints in class hierarchies, 655–658
- dynamic, 618–619
- function types and, 1147–1148
- text-segment size comparison, 1108
- type system and, 1089n5
- unconditional, 1085–1089
- violating, 1093

exceptions, 615–618, 1104
excess *N* notation, 155
executable images, 1135
execution character sets, 844
expansion. *See* pack expansion
expiring objects, 741–742, 749
expiring value
- rvalue references, 712–713
- *xvalues*, 721–723

explicit class APIs, 38–39
explicit constructors, passing multiple arguments, 250–252
explicit conversion operators
- description of, 61–63
- potential pitfalls, 66–67
- use cases, 63–65

explicit instantiation declarations
- annoyances, 373–375
  - member validity, 374–375
  - unrelated class definitions, 373–374
- description of, 353–365
- further reading for, 376
- illustrative example, 355–359
- .o files, effect on, 359–365
- potential pitfalls, 371–373
  - corresponding explicit-instantiation declarations and definitions, 371–372
  - pessimization over optimization, 373
- use cases, 365–370
  - insulation from client code, 369–370
  - reducing code bloat in object files, 365–369

explicit instantiation definitions, 353–355, 358–359, 363, 370–375
explicit instantiation directives, 353n1, 354–355, 369, 375
explicit template argument specifications, 895
explicit typenames, 26–27
explicitly captured, 582–583
explicitly copied, 583
explicitly declaring special member functions, 33–34
exporting bitwise copies of PODs, 479–480
expression alias, 1146–1147
expression SFINAE, 29n3, 122, 126
expression templates, 202–203
expressions. *See also* lambda expressions
- compound, **noexcept** operator and, 626–627
- **decltype** use with, 25–26
- decomposing complex, 391–393
- rvalue references in, 730–731
- validation of, 28–30

extended alignment, 168–170
extended **friend** declarations. *See* **friend** declarations
extended **typedef**. *See* aliases
extern template
- annoyances, 373–375
  - member validity, 374–375
  - unrelated class definitions, 373–374
- description of, 353–365
- further reading for, 376
- illustrative example, 355–359
- .o files, effect on, 359–365
- potential pitfalls, 371–373
  - corresponding explicit-instantiation declarations and definitions, 371–372
  - pessimization over optimization, 373
- use cases, 365–370
  - insulation from client code, 369–370
  - reducing code bloat in object files, 365–369

external definitions for static member variables, 314–315
external linkages, 307
external static analysis, control of, 17–18