

# Programación Orientada a Objetos

```
var OwlCarousel = {  
  init : function(options, el){  
    var base = this;  
  
    base.$elem = $(el);  
  
    // options passed via js override options passed via data attributes  
    base.options = $.extend({}, $.fn.owlCarousel.options, base.$elem.data(), options);  
  
    base.userOptions = options;  
    base.loadContent();  
  },  
  
  loadContent : function(){  
    var base = this;  
  
    if (typeof base.options.beforeInit === "function") {  
      base.options.beforeInit.apply(this, [base.$elem]);  
    }  
  
    if (typeof base.options.jsonPath === "string") {  
      var url = base.options.jsonPath;  
  
      function getData(data) {  
        if (base.options.jsonSuccess === "function") {  
          base.options.jsonSuccess.apply(this, [data]);  
        }  
      }  
    }  
  }  
};
```



## Tener en cuenta...

- Ciclo 3 – Proyecto Inicial  
Sábado 01 de Octubre
- Laboratorio 3  
Viernes 30 de Septiembre
- Práctica XP

## Coding

- The customer is always available.
- Code must be written to agreed standards.
- Code the unit test first.
- All production code is pair programmed.
- Only one pair integrates code at a time.
- Integrate often.
- Set up a dedicated integration computer.
- Use collective ownership.

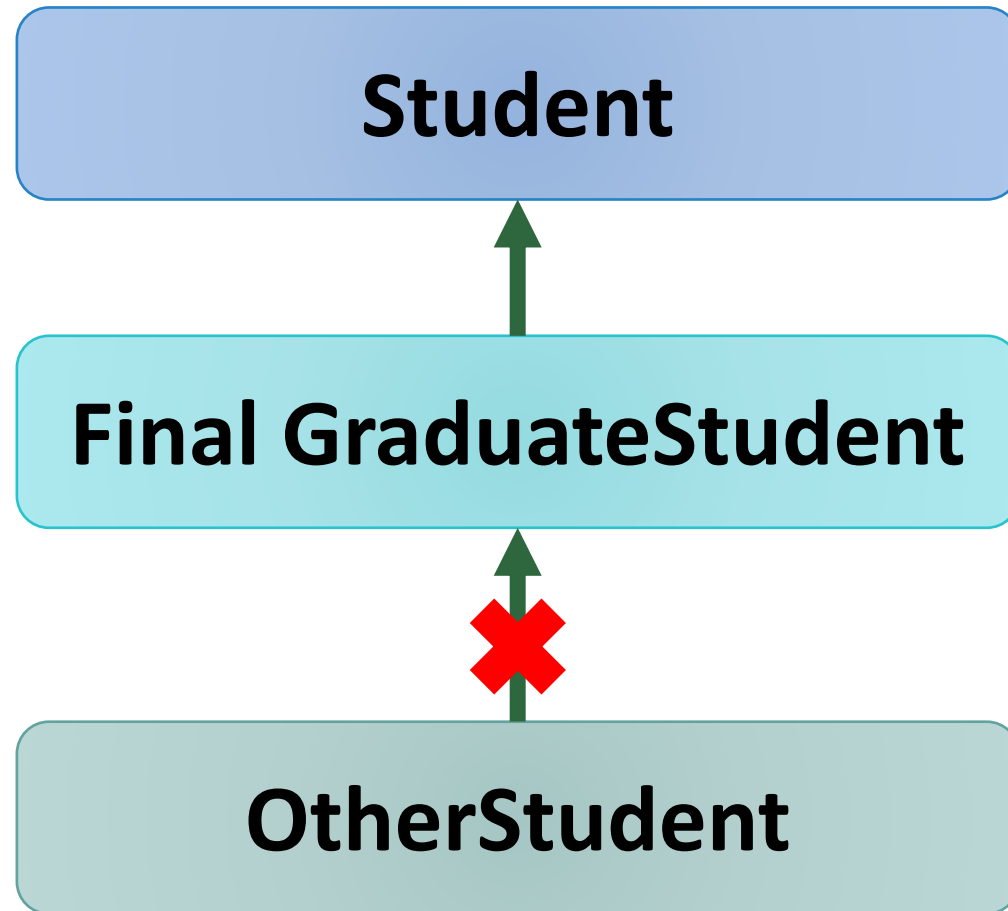
## Prácticas XP

- Exposición 5 minutos de:
  - ✓ ¿Qué propone?
  - ✓ ¿Para qué se utiliza?
  - ✓ ¿Cómo la usarían en POOB?

¿Voluntario 1?

¿Voluntario 2?

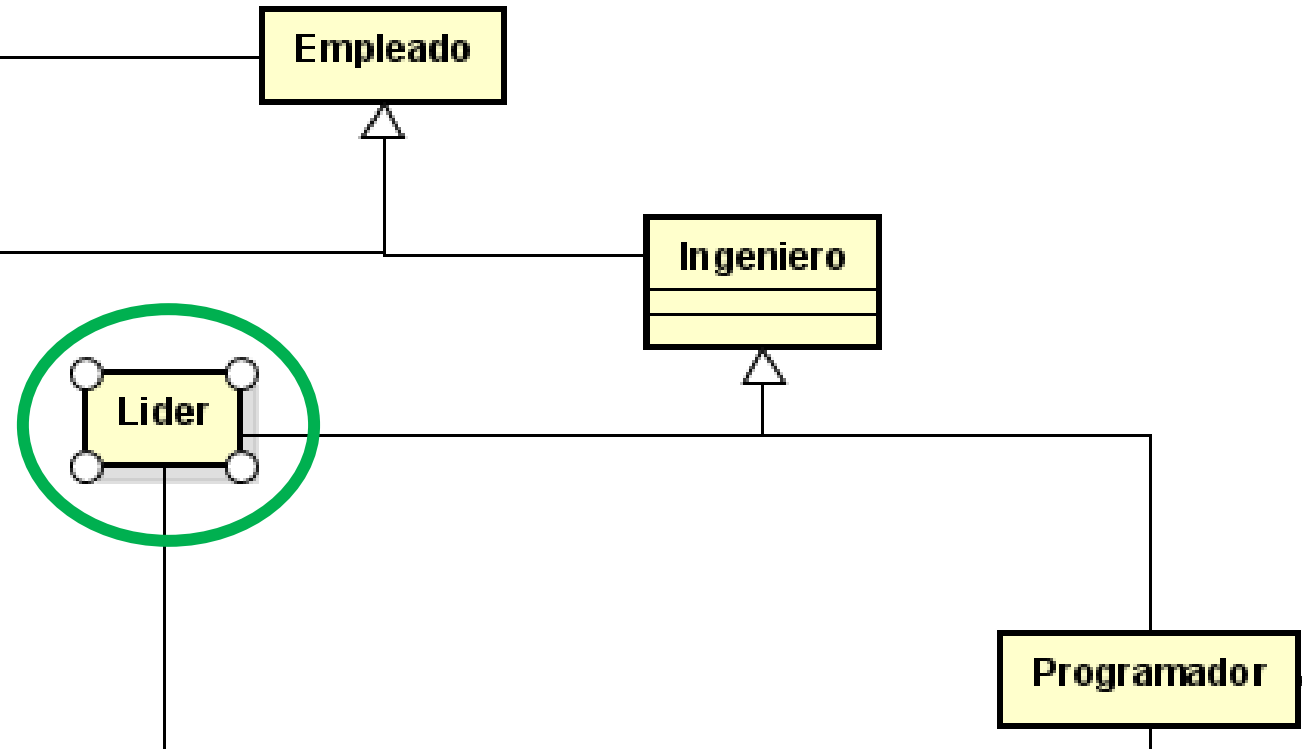
# Clase final



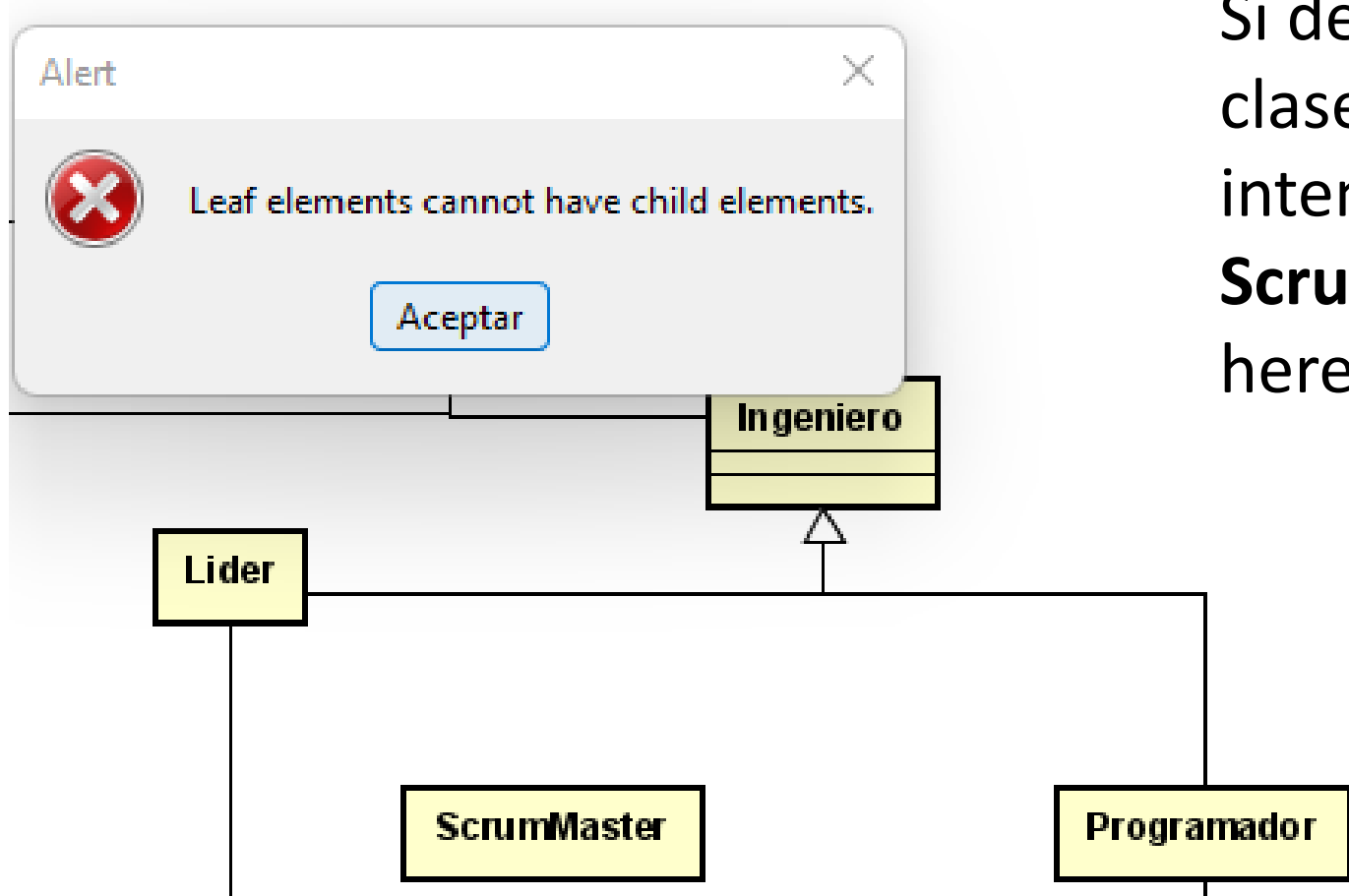
- Una clase final **no** puede ser heredada
- Una clase final -> Hoja de un árbol
- Máximo nivel de especialización

# Clase final

Template	Parameter	Constraint	Language	Hyperlink
Generalization	Dependency	Association	Property	
Base	Stereotype	Attribute	Operation	
Namespace				
Name	Lider			
Visibility	public			
Abstract	false			
Leaf	true			
Active	false			
Definition				



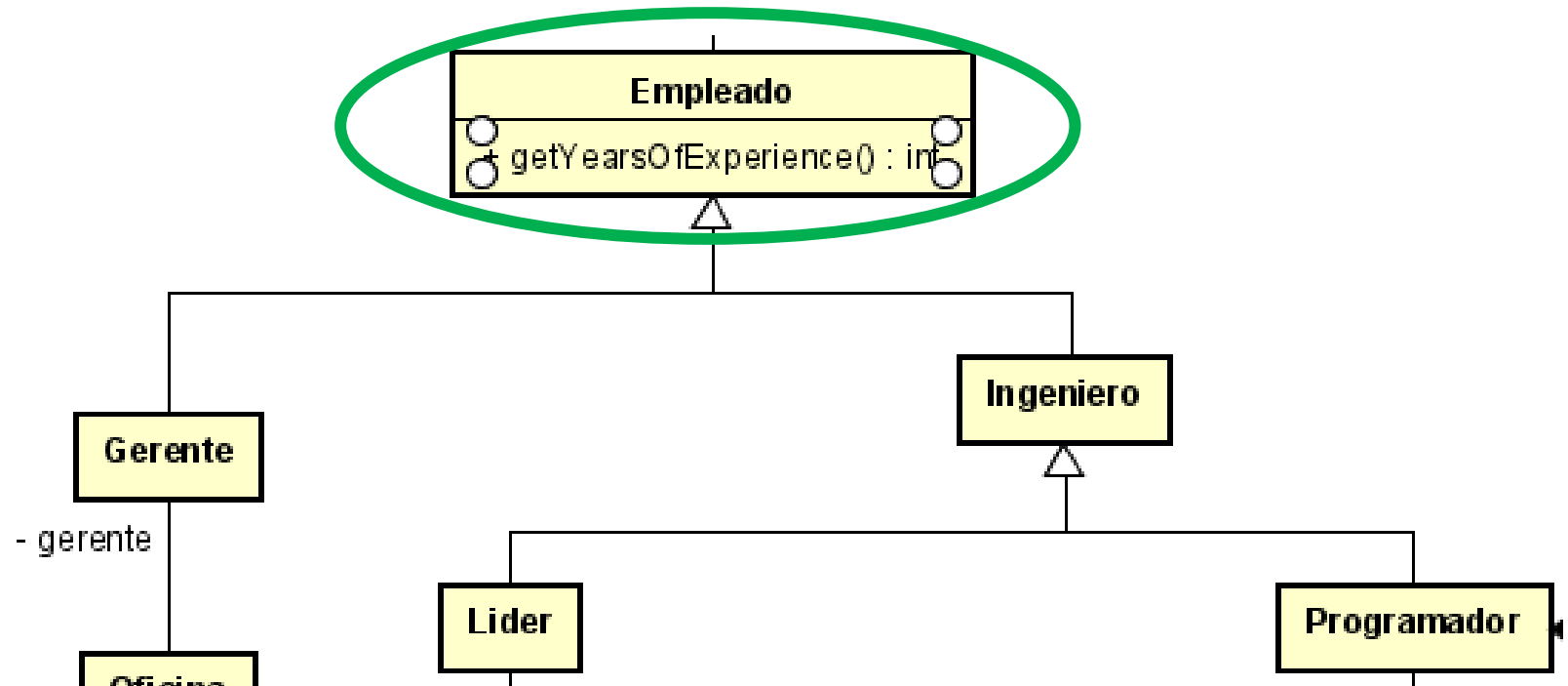
# Clase final



Si después de marcar la clase **Líder** como *final* intento que la clase **ScrumMaster** la herede

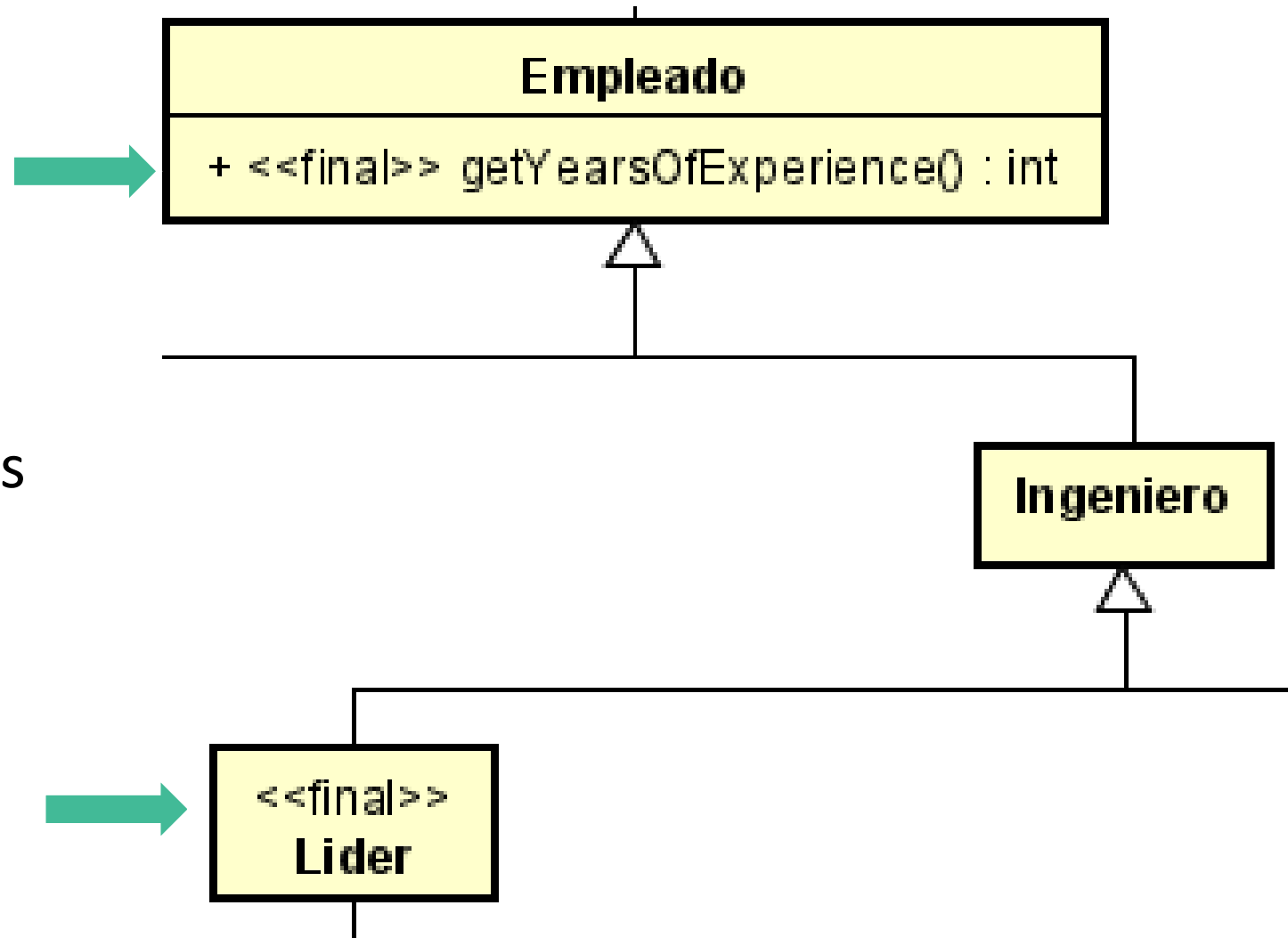
# Método final

Constraint	Language	Hyperlink
Post Condition	Body Condition	
Base	Parameters	Stereotype
Precondition		
Name	getYearsOfExperience	
Return Type	int	
Type Modifier		
Visibility	public	
Static	false	
Abstract	false	
Leaf	true	
Definition		



# Método y clase final

Usar *etiquetas (stereotype)* para hacer más visible si es una clase o método *final*





# Clase abstracta

Generaliza comportamientos y características, **pero no tiene sentido instanciar objetos de ella**

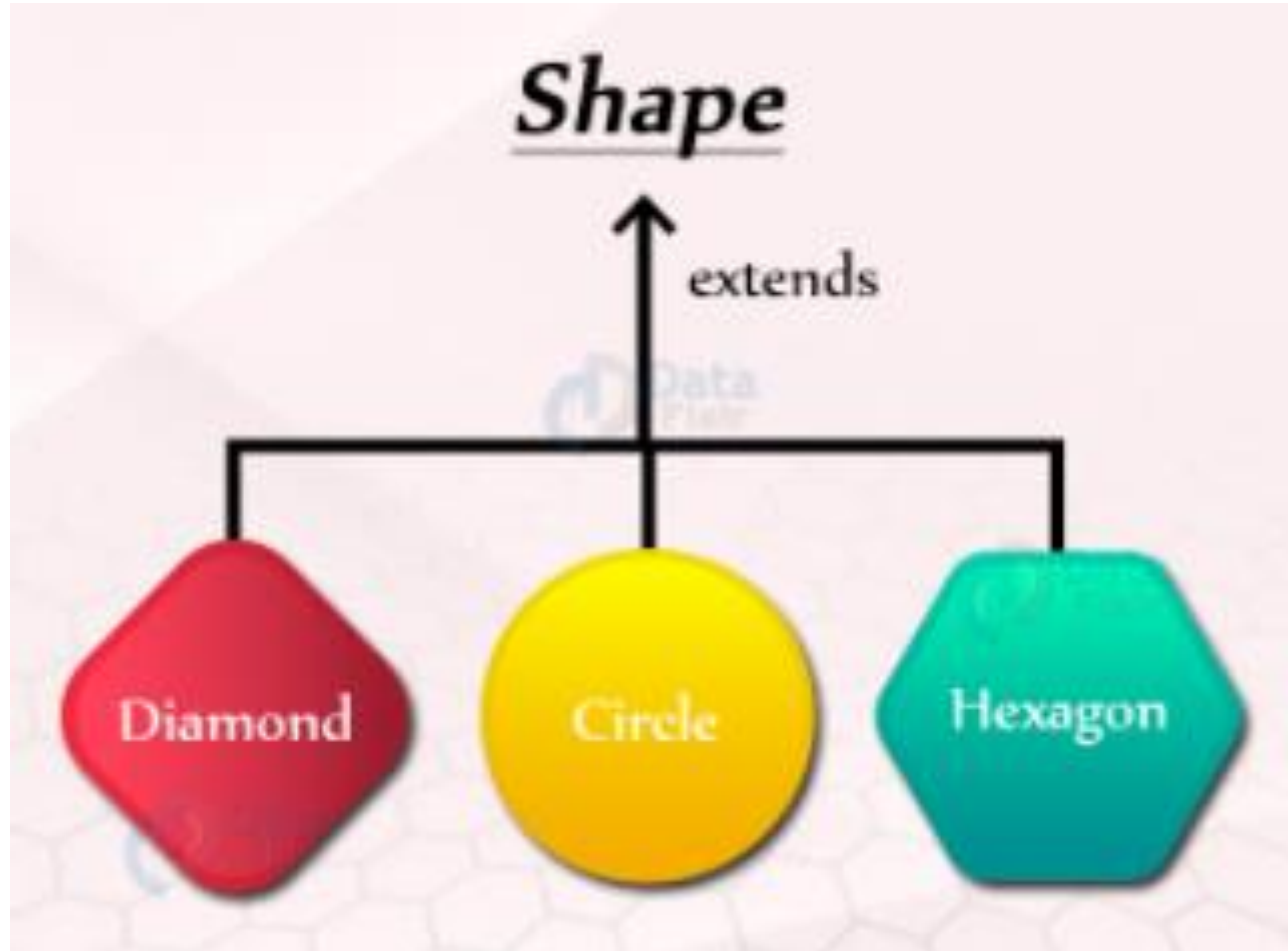


Imagen tomada de: <https://data-flair.training/blogs/abstract-class-in-java/>

# Método abstracto

El comportamiento es **generalizado**, pero su **implementación** es **específica** de cada subclase

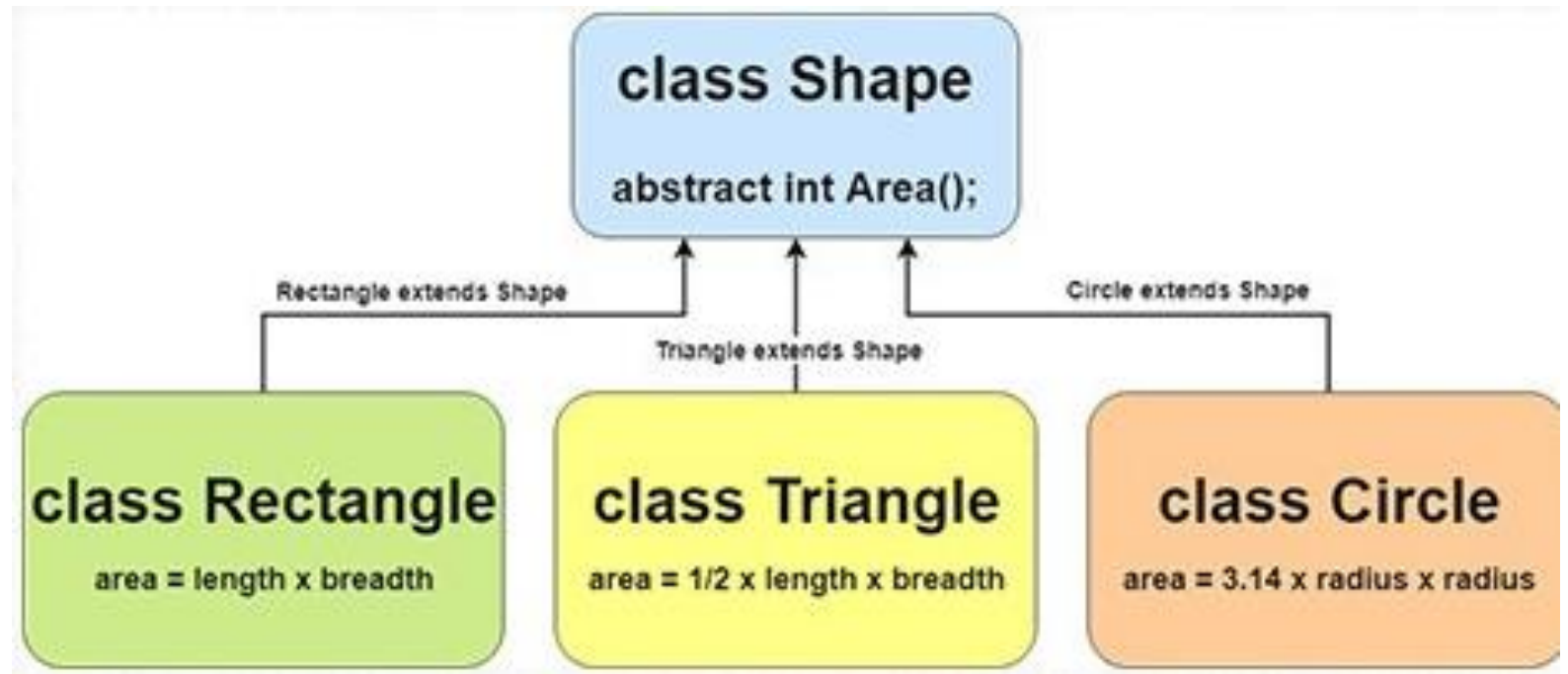


Imagen tomada de: <https://simplesnippets.tech/abstract-class-abstract-methods-in-java/>

# Clase abstracta

- Clase= Métodos + atributos.



Debe tener mínimo un método abstracto

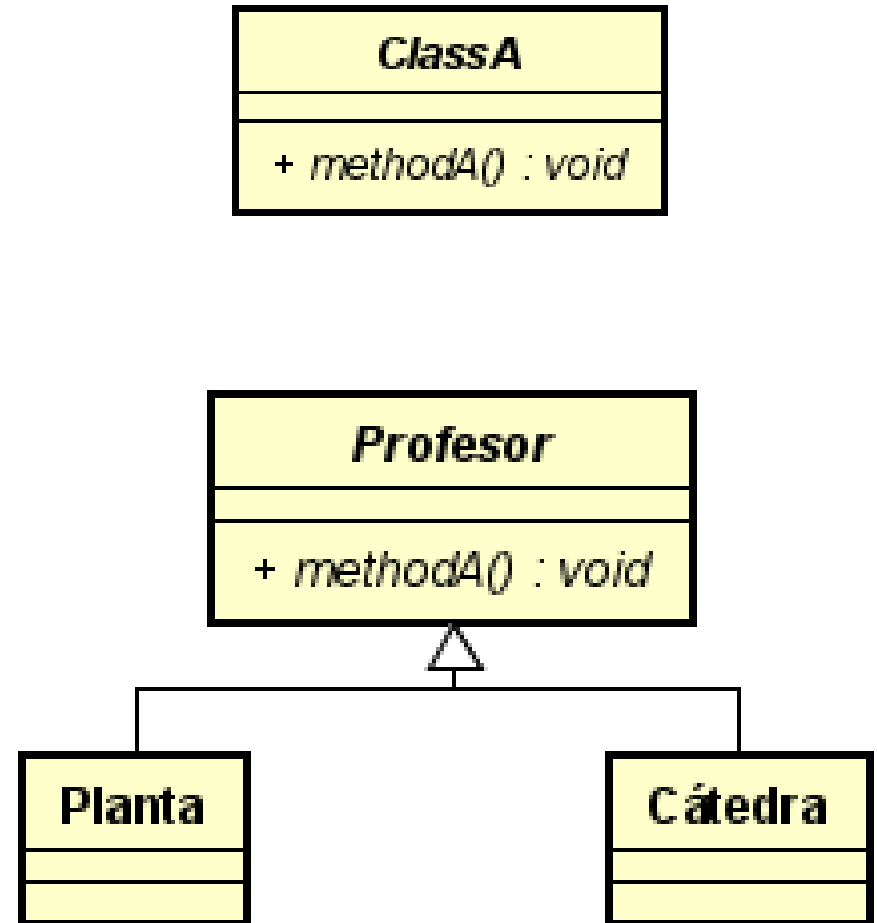


Método declarado pero no implementado

- No puede ser instanciada:



ClassA classA = new ClassA();



# Interfaz

Es una **plantilla**

Generaliza  
comportamientos **sin  
decir el cómo**

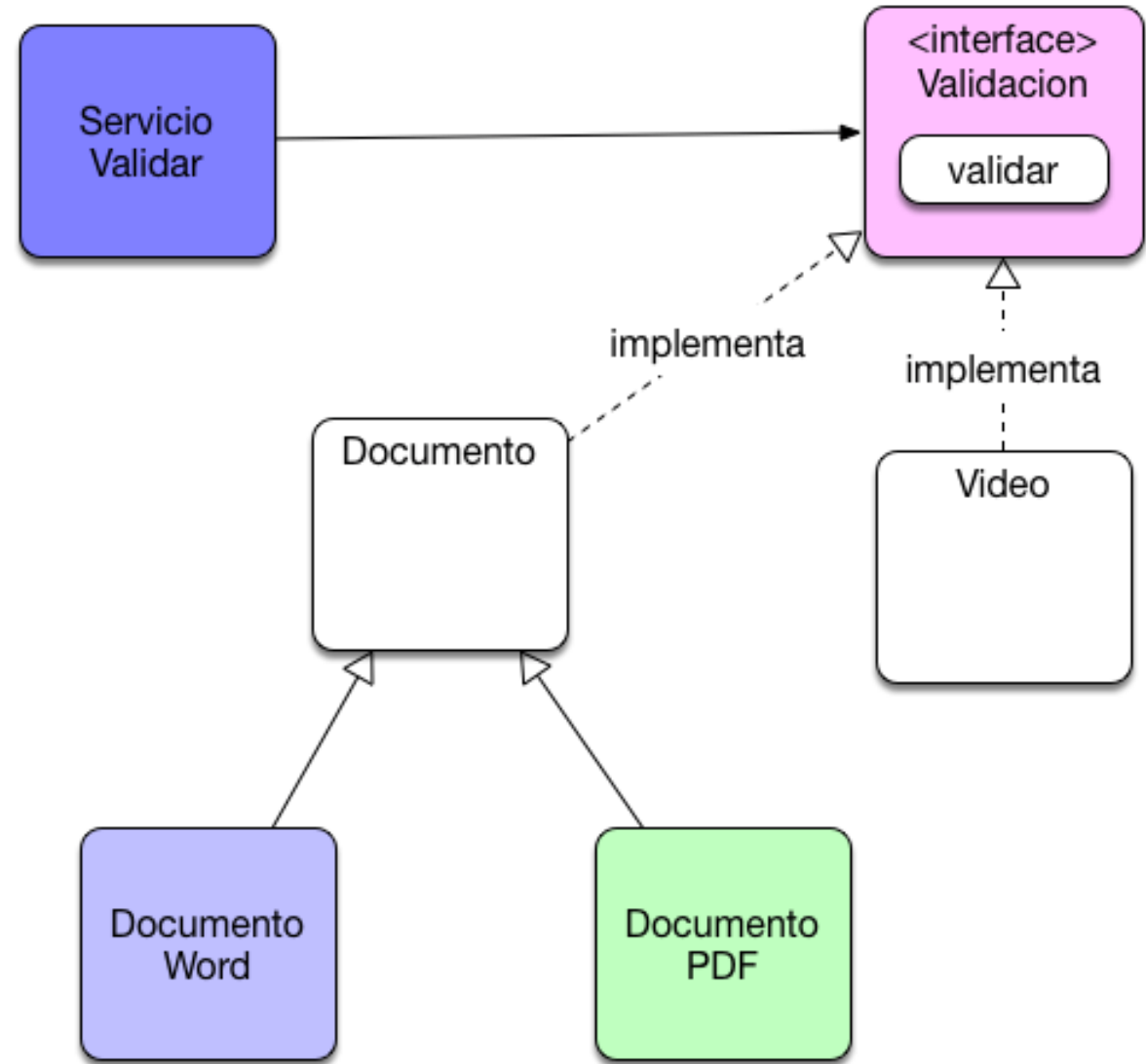


Imagen tomada de: <https://www.arquitecturajava.com/java-herencia-vs-interfaces/>

# Interfaz


- Clase = Métodos - Atributos.

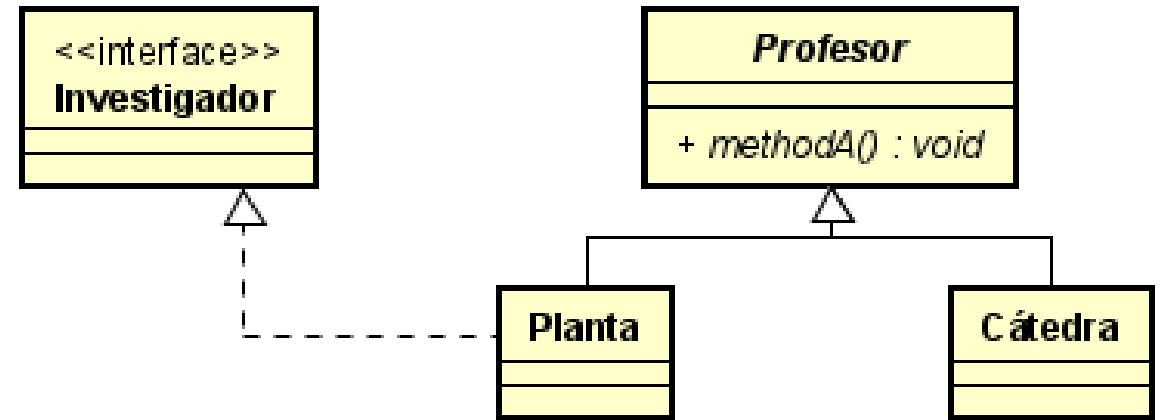


Plantilla: Encabezados



Métodos *default*

- No puede ser instanciada:  
 Investigador = new Investigador();
- Herencia múltiple

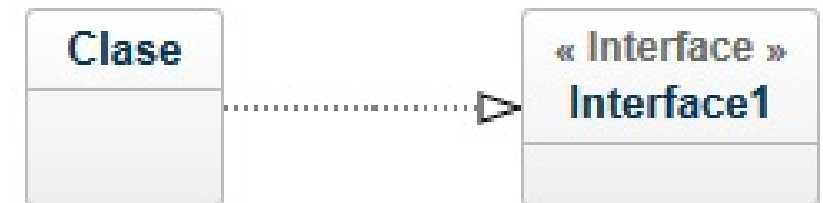


# Interfaz – Java 8

- Se permite implementación de métodos: Métodos *default*. -> Las clases que implementen la interfaz no están obligadas a escribir su código.

**Objetivo:** Si se modifica una interfaz añadiendo una funcionalidad, evita tener que modificar el código para todas las demás clases que la implementan.

- Puede contener métodos privados.



# Palabras reservadas

Palabra	Nivel	Objetivo
Extends	Clase	Herencia
Final	Clase	Nivel máximo de especialización
Final	Método	No se permite sobreescritura
Abstract	Clase	No se puede instanciar
Abstract	Método	Las subclases tienen que tener implementado el método
Interface	Clase	Es abstracta y establecen la forma que debe tener una clase. Define qué y no el cómo *.
Implements	Clase	Una clase implementa una interfaz

# Pruebas de Software

## TDD





# Pruebas de Software

```
import static org.junit.Assert.*;
import org.junit.Test;

public class TestLogic {

    @Test
    public void testFindMax(){
        assertEquals(4, Calculation.findMax(new int[]{1,3,4,2}));
        assertEquals(-1, Calculation.findMax(new int[]{-12,-1,-3,-4,-2}));
    }
}
```

JUnit 

## ¿QUÉ TIPOS DE TESTING EXISTEN?



Son pruebas que permiten asegurar la calidad y funcionalidad del producto que estamos desarrollando.



### PRUEBAS UNITARIAS (UNIT TESTING)

Son funciones cuyo objetivo es **probar componentes específicos**, en otras palabras, es **código que se crea para probar código**.

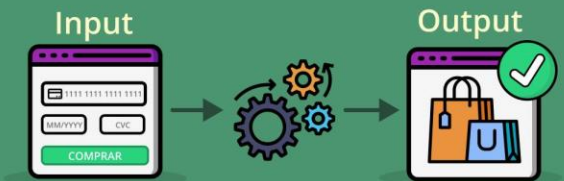


### PRUEBAS DE INTEGRACIÓN

Verifican que los **distintos componentes** de un sistema, **funcionan juntos** correctamente.

### PRUEBAS END TO END (E2E)

Simulamos la **interacción** que hace el **usuario con la aplicación**, ingresando datos al sistema (**input**) y obteniendo una salida (**output**).



Tomado de: <https://ed.team/cursos/testing>

# Pruebas de Software

## Methods of Assert class

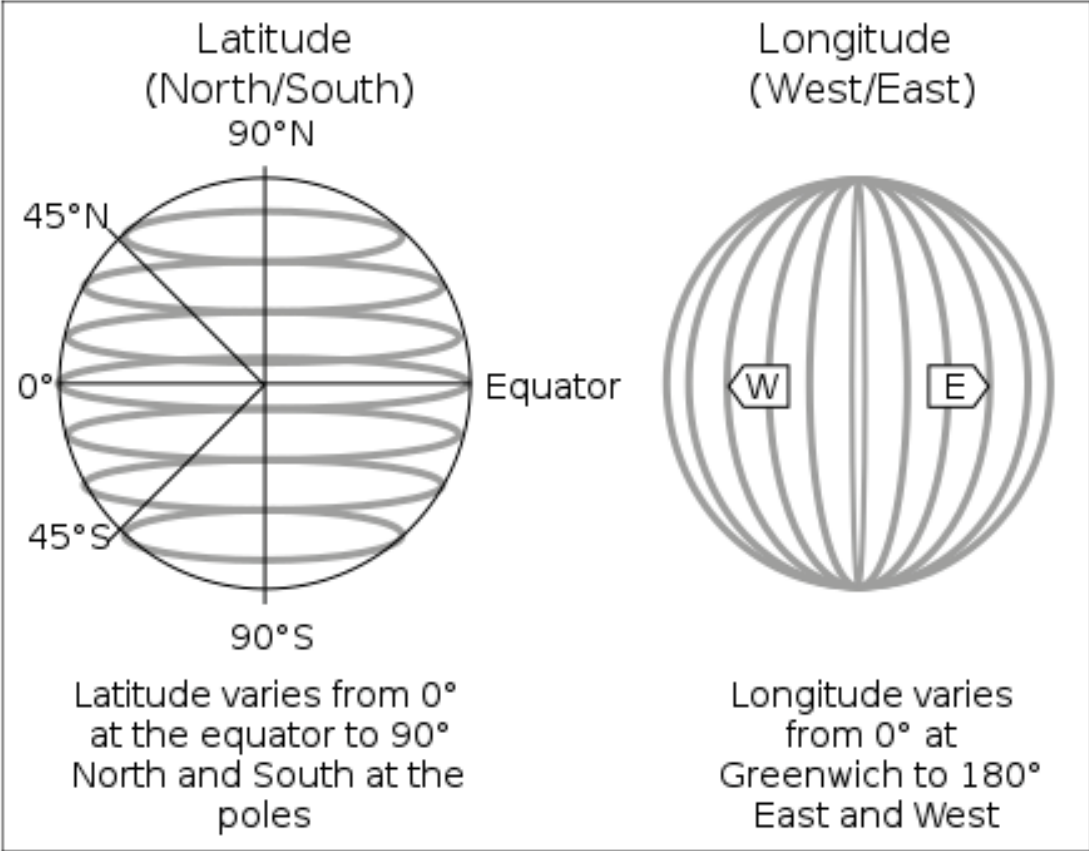
The common methods of Assert class are as follows:

1. **void assertEquals(boolean expected,boolean actual)**: checks that two primitives/objects are equal. It is overloaded.
2. **void assertTrue(boolean condition)**: checks that a condition is true.
3. **void assertFalse(boolean condition)**: checks that a condition is false.
4. **void assertNull(Object obj)**: checks that object is null.
5. **void assertNotNull(Object obj)**: checks that object is not null.

# File:FedStats Lat long.svg

From Wikipedia, the free encyclopedia

	File	File history
--	------	--------------



<code>void</code>	<code>alNorte()</code> Mueve las máquinas de la flota una posición al norte. El mundo tablero es circular. Longitud [0,180] Latitud [-90,90]. <a href="#">Coordenadas.</a>
<code>ArrayList&lt;Maquina&gt;</code>	<code>seranDestruidas(int longitud, int latitud)</code> Consulta las máquinas que pueden afectarse por una explosión en la posición dada. En una coordenada pueden estar muchas máquinas. Los aviones en aire no se destruyen.
<code>ArrayList&lt;Maquina&gt;</code>	<code>maquinasDebiles()</code> Consulta las máquinas débiles de una flota Un barco es débil si tienen menos de cinco marinos; un avión, si no tiene piloto principal; y un portaaviones si es un barco débil o alguno de sus aviones en aire es débil. <b>Returns:</b> Las máquinas débiles
<code>void</code>	<code>ataquen(int lon, int lat)</code> Mueve todas las máquinas que no son débiles paso a paso (uno a uno) hacia la posición a atacar indicada por (lon, lat) <b>Parameters:</b> lon - longitud lat - latitud

Size of this PNG-rendered file: 405 x 340 pixels. Other resolutions: 307 x 254 pixels, 1024 x 840 pixels, 1678 x 1382 pixels, 2048 x 1678 pixels, 4096 x 3356 pixels, 8192 x 6712 pixels, 16384 x 13424 pixels, 32768 x 26848 pixels, 65536 x 53696 pixels, 131072 x 107392 pixels, 262144 x 214784 pixels, 524288 x 429568 pixels, 1048576 x 859136 pixels, 2097152 x 1718272 pixels, 4194304 x 3436544 pixels, 8388608 x 6873088 pixels, 16777216 x 13746176 pixels, 33554432 x 27492352 pixels, 67108864 x 54984704 pixels, 134217728 x 109969408 pixels, 268435456 x 219938816 pixels, 536870912 x 439877632 pixels, 1073741824 x 879755264 pixels, 2147483648 x 1759510528 pixels, 4294967296 x 3519021056 pixels, 8589934592 x 7038042112 pixels, 17179869184 x 14076084224 pixels, 34359738368 x 28152168448 pixels, 68719476736 x 56304336896 pixels, 137438953472 x 112608673792 pixels, 274877906944 x 225217347584 pixels, 549755813888 x 450434695168 pixels, 1099511627776 x 900869390336 pixels, 2199023255552 x 1801738780672 pixels, 4398046511104 x 3603477561344 pixels, 8796093022208 x 7206955122688 pixels, 17592186044416 x 14413910245376 pixels, 35184372088832 x 28827820490752 pixels, 70368744177664 x 57655640981504 pixels, 140737488355328 x 115311281963008 pixels, 281474976710656 x 230622563926016 pixels, 562949953421312 x 461245127852032 pixels, 1125899906842624 x 922490255704064 pixels, 2251799813685248 x 1844980511408128 pixels, 4503599627370496 x 3689961022816256 pixels, 9007199254740992 x 7379922045632512 pixels, 18014398509481984 x 14759844091265024 pixels, 36028797018963968 x 29519688182530048 pixels, 72057594037927936 x 59039376365060096 pixels, 144115188075855872 x 118078752730120192 pixels, 288230376151711744 x 236157505460240384 pixels, 576460752303423488 x 472315010920480768 pixels, 1152921504606846976 x 944630021840961536 pixels, 2305843009213693952 x 1889260043681923072 pixels, 4611686018427387904 x 3778520087363846144 pixels, 9223372036854775808 x 7557040174727692288 pixels, 18446744073709551616 x 15114080349455384576 pixels, 36893488147419103232 x 30228160698910769152 pixels, 73786976294838206464 x 60456321397821538304 pixels, 147573952589676412928 x 120912642795643076608 pixels, 295147905179352825856 x 241825285591286153216 pixels, 590295810358705651712 x 483650571182572306432 pixels, 1180591620717411303424 x 967301142365144612864 pixels, 2361183241434822606848 x 1934602284730289225728 pixels, 4722366482869645213696 x 3869204569460578451456 pixels, 9444732965739290427392 x 7738409138921156902912 pixels, 18889465931478580854784 x 15476818277842313805824 pixels, 37778931862957161709568 x 30953636555684627611648 pixels, 75557863725914323419136 x 61907273111369255223296 pixels, 151115727451828646838272 x 123814546222738510446592 pixels, 302231454903657293676544 x 247629092445477020893184 pixels, 604462909807314587353088 x 495258184890954041786368 pixels, 1208925819614629174706176 x 990516369781908083572736 pixels, 2417851639229258349412352 x 1981032739563816167145472 pixels, 4835703278458516698824704 x 3962065479127632334290944 pixels, 9671406556917033397649408 x 7924130958255264668581888 pixels, 19342813113834066795298816 x 15848261916510529337163776 pixels, 38685626227668133590597632 x 31696523833021058674327552 pixels, 77371252455336267181195264 x 63393047666042117348655104 pixels, 154742504910672534362390528 x 126786095332084234697310208 pixels, 309485009821345068724781056 x 253572190664168469394620416 pixels, 618970019642690137449562112 x 507144381328336938789240832 pixels, 1237940039285380274899124224 x 1014288762656673877578481664 pixels, 2475880078570760549798248448 x 2028577525313347755156963328 pixels, 4951760157141521099596496896 x 4057155050626695510313926656 pixels, 9903520314283042199192993792 x 8114310101253391020627853312 pixels, 19807040628566084398385987584 x 16228620202506782041255706624 pixels, 39614081257132168796771975168 x 32457240405013564082511413248 pixels, 79228162514264337593543950336 x 64914480810027128165022826496 pixels, 158456325028528675187087900672 x 129828961620054256330045652992 pixels, 316912650057057350374175801344 x 259657923240108512660091305984 pixels, 633825300114114700748351602688 x 519315846480217025320182611968 pixels, 1267650600228229401496703205376 x 1038631692960434050640365223936 pixels, 2535301200456458802993406410752 x 2077263385920868101280730447872 pixels, 5070602400912917605986812821504 x 4154526771841736202561460895744 pixels, 10141204801825835211973625643008 x 8309053543683472405122921791488 pixels, 20282409603651670423947251286016 x 16618107087366944810245843582976 pixels, 40564819207303340847894502572032 x 33236214174733889620491687165952 pixels, 81129638414606681695789005144064 x 66472428349467779240983374331904 pixels, 162259276829213363391578010288128 x 132944856698935558481966748663808 pixels, 324518553658426726783156020576256 x 265889713397871116963933497327616 pixels, 649037107316853453566312041152512 x 531779426795742233927866994655232 pixels, 1298074214633706907132624082305024 x 1063558853591484467855733989310464 pixels, 2596148429267413814265248164610048 x 2127117707182968935711467978620928 pixels, 5192296858534827628530496329220096 x 4254235414365937871422935957241856 pixels, 10384593717069655257060992658440192 x 8508470828731875742845871914483712 pixels, 20769187434139310514121985316880384 x 17016941657463751485691743828967424 pixels, 41538374868278621028243970633760768 x 34033883314927502971383487657934848 pixels, 83076749736557242056487941267521536 x 68067766629855005942766975315869696 pixels, 166153499473114484112975882535043072 x 136135533259710011885533950631739392 pixels, 332306998946228968225951765070086144 x 272271066519420023771067901263478784 pixels, 664613997892457936451903530140172288 x 544542133038840047542135802526957568 pixels, 1329227995784915872903807060280344576 x 1089084266077680095084271605053915136 pixels, 2658455991569831745807614120560689152 x 2178168532155360190168543210107830272 pixels, 5316911983139663491615228241121378304 x 4356337064310720380337086420215660544 pixels, 10633823966279326983230456482242756608 x 8712674128621440760674172840431321088 pixels, 21267647932558653966460912964485513216 x 17425348257242881521348345680862642176 pixels, 42535295865117307932921825928971026432 x 34850696514485763042696691361725284352 pixels, 85070591730234615865843651857942052864 x 69701393028971526085393382723450568704 pixels, 170141183460469231731687303715884105728 x 139402786057943052170786765446901137408 pixels, 340282366920938463463374607431768211456 x 278805572115886104341573530893802274816 pixels, 680564733841876926926749214863536422912 x 557611144231772208683147061787604549632 pixels, 1361129467683753853853498429727072845824 x 1115222288463544417366294123575209099264 pixels, 2722258935367507707706996859454145691648 x 2230444576927088834732588247150418198528 pixels, 5444517870735015415413993718908291383296 x 4460889153854177669465176494300836397056 pixels, 10889035741470030830827987437816582766592 x 8921778307708355338930352988601672794112 pixels, 21778071482940061661655974875633165533184 x 17843556615416710677860705977203345588224 pixels, 43556142965880123323311949751266331066368 x 35687113230833421355721411954406691176448 pixels, 87112285931760246646623899502532662132736 x 71374226461666842711442823908813382352896 pixels, 174224571863520493293247799005065324265472 x 142748452923333685422885647817626764705792 pixels, 348449143727040986586495598010130648530944 x 285496905846667370845771295635253529411584 pixels, 696898287454081973172991196020261297061888 x 570993811693334741691542591270507058823168 pixels, 1393796574908163946345982392040522594123776 x 1141987623386669483383085182541014117646336 pixels, 2787593149816327892691964784081045188247552 x 2283975246773338966766170365082028235292672 pixels, 5575186299632655785383929568162090376495104 x 4567950493546677933532340730164056470585344 pixels, 11150372599265311570767859136324180752990208 x 9135900987093355867064681460328112941170688 pixels, 22300745198530623141535718272648361505980416 x 18271801974186711734129362920656225882341376 pixels, 44601490397061246283071436545296723011960832 x 36543603948373423468258725841312451764682752 pixels, 89202980794122492566142873090593446023921664 x 73087207896746846936517451682624903529365504 pixels, 178405961588244985132285746181186892047843328 x 146174415793493693873034903365249807058731008 pixels, 356811923176489970264571492362373784095686656 x 292348831586987387746069806730499614117462016 pixels, 713623846352979940529142984724747568191373312 x 584697663173974775492139613460999228234924032 pixels, 1427247692705959881058285969449495136382746624 x 1169395326347949550984279226921998456469848064 pixels, 2854495385411919762116571938898990272765492128 x 2338790652695899101968558453843996912939696128 pixels, 5708990770823839524233143877797980545530984256 x 4677581305391798203937116907687993825879392256 pixels, 11417981541647679048466287755595961091061968512 x 9355162610783596407874233815375987651758784512 pixels, 22835963083295358096932575511191922182123937024 x 18710325221567192815748467630751975303517569024 pixels, 45671926166590716193865151022383844364247874048 x 37420650443134385631496935261503950607035138048 pixels, 91343852333181432387730302044767688728495768096 x 74841300886268771262993870523007901214070276096 pixels, 182687704666362864775460604089535377456991536192 x 149682601772537542525987741046015802428140552192 pixels, 365375409332725729550921208179070754913983072384 x 299365203545075085051975482092031604856281104384 pixels, 730750818665451459101842416358141509827966144768 x 598730407090150170103950964184063209712562208768 pixels, 1461501637330902918203684832716283019655932389536 x 1197460814180300340207901928368126419425124417536 pixels, 2923003274661805836407369665432566039311864739072 x 2394921628360600680415803856736252838850248835072 pixels, 5846006549323611672814739330865132078623729478144 x 4789843256721201360831607713472505677700497670144 pixels, 11692013098647223345629478661730264157247458956288 x 9579686513442402721663215426945011355400995340288 pixels, 23384026197294446691258957323460528314494917912576 x 19159373026884805443326430853890022710801990680576 pixels, 46768052394588893382517914646921056628989835825152 x 38318746053769610886652861707780045421603981361152 pixels, 93536104789177786765035829293842113257979671650304 x 76637492107539221773305723415560090843207962722304 pixels, 187072209578355573530071658587684226515959343300608 x 153274984215078443546611446831120181686415925444608 pixels, 374144419156711147060143317175368453031918686601216 x 306549968430156887093222893662240363372831850889216 pixels, 748288838313422294120286634350736906063837373202432 x 613099936860313774186445787324480726745663701778432 pixels, 1496577676626844588240573268701473812127674746404864 x 1226199873720627548372891574648961453491327403556864 pixels, 2993155353253689176481146537402947624255348892809728 x 2452399747441255096745783149297922906982654807113728 pixels, 5986310706507378352962293074805895248510697785619456 x 4904799494882510193491566298595845813965309614227456 pixels, 11972621413014756705924586149611790497021395371238912 x 9809598989765020386983132597191691627930619228454912 pixels, 23945242826029513411849172299223580994042790742479824 x 19619197979530040773966265194383383255861238456909824 pixels, 47890485652059026823698344598446761988085476884959648 x 39238395959060081547932530388766766511722476913819648 pixels, 95780971304118053647396689196893523976170953769919296 x 78476791918120163095865060777533533023444953827639296 pixels, 191561942608236107294793378393787047952341907539838592 x 156953583836240326191730121555067066046889907655278592 pixels, 383123885216472214589586756787574095904683815079677184 x 313907167672480652383460243110134132093779815310557184 pixels, 766247770432944429179173513575148191809367630159354368 x 627814335344961304766920486220268264187559630621114368 pixels, 1532495540865888858358347027150296383618735260318708736 x 1255628670689922609533840972440536528375119261242228736 pixels, 3064991081731777716716694054300592767237470520637417472 x 2511257341379845219067681944881073056750238522484457472 pixels, 6129982163463555433433388108601185534474941041274834944 x 5022514682759690438135363889762146113500477044968914944 pixels, 12259964326927110866866776217202371068949882082549669888 x 10045029365519380876270727779524292227000954089937829888 pixels, 24519928653854221733733552434404742137899764165099339776 x 20090058731038761752541455559048584454001908179875659776 pixels, 49039857307708443467467104868809484275799528330198679552 x 40180117462077523505082911118097168908003816359751319552 pixels, 98079714615416886934934209737618968551599056660397359104 x 80360234924155047010165822236194337816007632719502639104 pixels, 196159429230833773869868419475237937103198113320794718208 x 160720469848310094020331644472388675632015265439005278208 pixels, 3923188584616675477397