In [1]:
```
from sympy import *
```

Example 1: Show the ODE 2x + y^2 + 2xyy' = 0 is exact and solve it

In [3]:
```
x,y=symbols('x y')
M=2*x+y**2
N=2*x*y
My=diff(M,y)
Nx=diff(N,x)
print('My=',My,'and Nx=',Nx,'so the equation is an exact equation.')
# Solving using dsolve
x=symbols('x')
y=Function('y')
deq=2*x+y(x)**2+2*x*y(x)*diff(y(x),x)
ysoln=dsolve(deq,y(x))
print('The solution to the ODE is',ysoln)  #NOTE Python solved explicitly for y(x)
```

```
My= 2*y and Nx= 2*y so the equation is an exact equation.
The solution to the ODE is [y(x) == -sqrt(C1/x - x), y(x) == sqrt(C1/x - x)]
```

Example 2: Determine whether the IVP dy/dt = (e^t *sin(y) - 3y) / (3t - e^t* cos(y)), y(0)=pi/4 is exact or not. If so, solve it.

NOTE we first rewrite the equation as (3y - e^t *sin(y)) + (3t - e^t* cos(y)) y' = 0

In [5]:
```
t,y=symbols('t y')
M=3*y-exp(t)*sin(y)
N=3*t-exp(t)*cos(y)
My=diff(M,y)
Nt=diff(N,t)
print('My=',My,'and Nt=',Nt,'so the equation is an exact equation.')
# solve using dsolve
t=symbols('t')
y=Function('y')
deq=-(exp(t)*sin(y(t))-3*y(t))+(3*t-exp(t)*cos(y(t)))*diff(y(t),t)
ysoln=dsolve(deq,ics={y(0):pi/4})  # NOTE that we can tell Python which method to use to solve!
# a list of hints are available at https://docs.sympy.org/latest/modules/solvers/ode.html#hint-fun
ctions
print('The solution to the IVP is',ysoln)
```

```
My= -exp(t)*cos(y) + 3 and Nt= -exp(t)*cos(y) + 3 so the equation is an exact equation.
The solution to the IVP is 3*t*y(t) - exp(t)*sin(y(t)) == C1
```

Example 3: Show that the ODE y + (2xy - e^(-2y))y' = 0 is not exact, but that there is an integrating factor. Then solve the ODE.

In [13]:
```
# We will just use dsolve here
x=symbols('x')
y=Function('y')
deq=y(x)+(2*x*y(x)-exp(-2*y(x)))*diff(y(x),x)
ysoln=dsolve(deq,hint='1st_exact')
print('The solution to the ODE is',ysoln)
```

```
The solution to the ODE is x*exp(2*y(x)) - log(y(x)) == C1
```

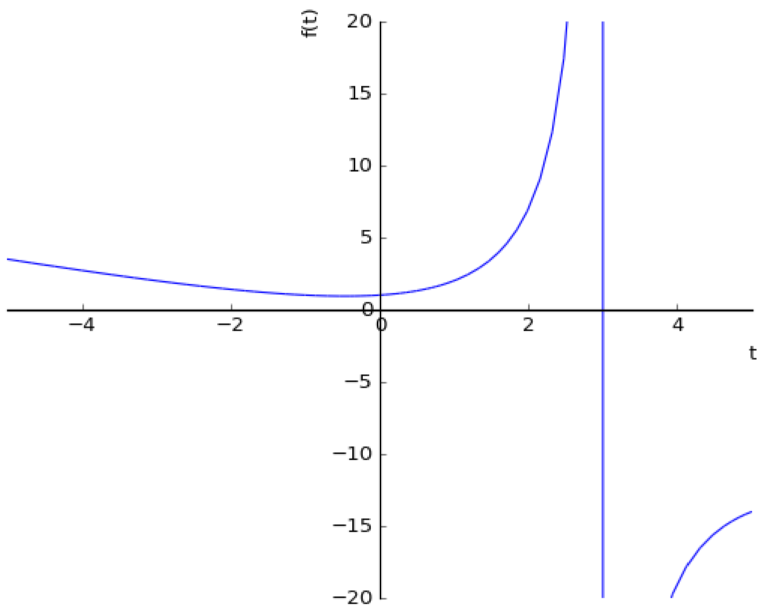
Example 4: Show that the ODE (-y^2-2ty)+(3+t^2)y' = 0 is not exact, but that there is an integrating factor. Then solve the ODE.

In [2]:
```
# We will just use dsolve here
t=symbols('t')
y=Function('y')
deq=(-y(t)**2-2*t*y(t))+(3+t**2)*diff(y(t),t)
ysoln=dsolve(deq,hint='1st_exact')
print('The solution to the ODE is',ysoln)  #solved explicitly for y
```

```
The solution to the ODE is y(t) == -(t**2 + 3)/(C1 + t)
```

In [7]:
```
matplotlib notebook
```

In [11]:
```
# ONE EXTRA EXAMPLE: Plot the solution to the IVP with y(0)=1 in domain [-5,5] and range [-20,20]
ysoln=dsolve(deq,y(t),ics={y(0):1})
yoft=-(t**2+3)/(-3+t)
# NOTE: I am using an older version of Python where ics does not work, so just typing the function
in here.
# You obviously would use the .rhs option to get the function!
plot(yoft,(t,-5,5),ylim=[-20,20])
```



Out[11]: <sympy.plotting.plot.Plot at 0x7f4319a42a20>

In [ ]: